



**Tiago
José Pires
Oliveira**

Ofertas de Emprego Mobile





**Tiago
José Pires
Oliveira**

Ofertas de Emprego Mobile

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Doutor Joaquim Arnaldo Carvalho Martins, Professor Catedrático do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e por Filipe Miguel Ferreira Dos Santos Augusto, Engenheiro da empresa Maisis - Information Systems, Lda.

o júri / the jury

presidente / president

Prof. Doutora Ana Maria Perfeito Tomé

Professora Associada, Universidade de Aveiro

vogais / examiners committee

Prof. Doutor Fernando Joaquim Lopes Moreira

Professor Associado, Departamento de Inovação, Ciência e Tecnologia da Universidade Portucalense (Arguente Principal)

Prof. Doutor Joaquim Arnaldo Carvalho Martins

Professor Catedrático, Universidade de Aveiro (Orientador)

agradecimentos

Em primeiro lugar, gostaria de agradecer ao meu orientador Professor Doutor Joaquim Arnaldo Carvalho Martins, pelo seu conhecimento, disponibilidade e interesse que foram cruciais para a realização desta dissertação.

Queria também manifestar o meu apreço à empresa Maisis - Information Systems, Lda. que me recebeu de braços abertos e me proporcionou a oportunidade de realizar o meu projeto. Em especial, agradeço ao meu orientador na empresa Eng. Filipe Augusto e a toda a equipa, que sempre demonstraram a disponibilidade e o apoio necessários para o sucesso do projeto.

Agradeço ainda a todas as pessoas que, no decorrer deste ano, estiveram ao meu lado e me ajudaram a superar todos os obstáculos que foram aparecendo. Em especial à minha família e aos meus amigos que sempre souberam dar o empurrão necessário quando as coisas se tornavam mais complicadas, e que fizeram questão de me acompanhar nesta caminhada.

palavras-chave

Ofertas de emprego, Mobile Development, Cross-platform, Frameworks, PhoneGap, Xamarin, MoSync, jQuery Mobile, Sencha Touch, AngularJS

resumo

No atual contexto de dificuldades ao nível do mercado de trabalho é fundamental apostar em medidas que contribuam para viabilizar e acelerar a procura de emprego. Com a crescente competitividade ao nível do mercado global torna-se fundamental o acesso às ofertas de emprego que melhor se adequam ao perfil de cada candidato, permitindo também ao mesmo ter uma perceção real que quais as capacidades que deve desenvolver para ter acesso a um maior e melhor conjunto de ofertas de emprego.

Os dispositivos móveis são atualmente objetos do quotidiano de um cada vez maior número de pessoas e tornaram-se num importante veículo no consumo de informação. Os principais concorrentes no mercado dos sistemas operativos móveis, têm nas suas lojas de aplicações uma oferta diversificada para as mais diferentes áreas, sendo o desenvolvimento destas aplicações uma área em forte expansão. A concorrência nas tecnologias de informação é normalmente marcada por uma diferenciação nas abordagens e cada um dos concorrentes de mercado aposta nas suas tecnologias. No sentido da confluência do desenvolvimento de aplicações móveis surgiu uma vertente de desenvolvimento usando tecnologias web tais como o HTML5, JavaScript e CSS3, permitindo a transversalidade às diferentes plataformas.

Conciliando estes dois cenários o objeto de estudo desta dissertação centrou-se no desenvolvimento de uma aplicação móvel para duas plataformas móveis, usando um modelo híbrido nativo-web, em que fosse possível através do perfil de um recurso humano obter quais as melhores ofertas de mercado de trabalho para esse recurso, facilitando dessa forma a procura de emprego. A aplicação desenvolvida neste contexto está disponível nas plataformas Android e IOS e tem já um animador conjunto de utilizadores.

abstract

In the current context of difficulties in the labor market is crucial to bet on measures to help facilitate and accelerate the job searching. With the increasing competitiveness in the global market, the access to job opportunities best suited to each candidate is a fundamental key, allowing the candidates to have a real perception of what capabilities they must develop to have access to a bigger and better set of jobs.

The mobile devices are now objects of everyday life for an increasing number of people and have become an important vehicle for the consumption of information. The main competitors in the market for mobile operating systems have in its stores a wide choice of applications for many different areas, being the development of these applications an expanding area. The competition in information technology is usually marked by a differentiation of approach and each of the market competitors bets in their technologies. As a consequence of the development of mobile applications has emerged a shed using web technologies such as HTML5, CSS3 and JavaScript, allowing transversality to different platforms.

Combining these two scenarios the aim of this thesis was focused on the development of a mobile application for two mobile platforms, using a web-native hybrid model, in which it was possible, through a human resource profile, get the job offers on the labor market for that resource, thereby facilitating job searching. The application developed in this context is available in the Android and iOS platforms and has already an encouraging set of users.

Conteúdo

Conteúdo	i
Lista de Figuras	v
Lista de Tabelas	vii
Acrónimos	ix
1 Introdução	1
1.1 Estrutura da Dissertação	1
1.2 A Empresa	2
1.3 Descrição do Sistema	2
1.4 Motivação e Objetivos	3
1.5 Público Alvo	3
1.6 Funcionalidades Gerais	4
2 Estado da Arte	7
2.1 Nativo, Web App ou Híbrido	9
2.2 Cross-Platform - Ferramentas de desenvolvimento	12
2.2.1 Xamarin	12
2.2.2 MoSync	13
2.2.3 PhoneGap	13
2.3 Frameworks de desenvolvimento de aplicações móveis	17
2.3.1 jQuery Mobile	17
2.3.2 Sencha Touch	18
2.3.3 AngularJS	18
2.4 Aplicações Móveis de Ofertas de Emprego	19
2.4.1 Indeed Jobs	19
2.4.2 Trovit Emprego	20
2.4.3 Net-empregos	21
3 Metodologia e Requisitos	25
3.1 Metodologia	25
3.1.1 Papéis	26
3.1.2 Artefactos	27
3.1.3 Atividades	27
3.2 Definição de Requisitos	28

3.2.1	Atores e Pessoas	28
3.2.2	Utilizador	28
3.3	Requisitos Funcionais	31
3.3.1	Requisitos contas de utilizador	32
3.3.2	Requisitos gerais da aplicação	32
3.3.3	Requisitos da oferta de emprego	32
3.3.4	Requisitos da pesquisa	33
3.3.5	Requisitos da estatística	33
3.3.6	Requisitos do perfil	33
3.3.7	Requisitos de interface externa	33
3.3.8	Requisitos de performance	34
3.3.9	Requisitos do sistema	34
4	Arquitetura e desenho	35
4.1	Arquitetura Servidor	35
4.1.1	Plataforma JOObian	35
4.1.2	Plataforma Funcional e Lógica do JOObian	37
4.1.3	Perspetiva Física	40
4.2	Arquitetura PhoneGap	43
4.3	Estrutura da aplicação móvel	46
4.3.1	Camada de dados	46
4.3.2	Camada lógica	47
4.3.3	Camada de representação	47
5	Desenvolvimento, Integração e Testes	49
5.1	Processo de desenvolvimento	50
5.1.1	Pré-requisitos	50
5.1.2	Serviços da plataforma JOObian	50
5.1.3	Criação de uma aplicação em PhoneGap	53
5.2	Desenvolvimento Web	55
5.2.1	Configurar e utilizar o AngularJS	56
5.2.2	Aplicação JOObian	58
5.2.3	Login	59
5.2.4	Registo	65
5.2.5	Pilha de Ofertas de Emprego	73
5.2.6	Favoritos	77
5.2.7	Perfil	78
5.2.8	Estatísticas	79
5.2.9	Definições	81
5.3	Testes	84
5.3.1	Testes de validação e aceitação	84
5.3.2	Testes realizados	84
5.3.3	Testes de compatibilidade e usabilidade	85

6	Conclusão	89
6.1	Projeto Desenvolvido	90
6.2	Apreciação pessoal	90
6.3	Trabalho Futuro	91
	Bibliografia	93
A	Documento de Requisitos	95
B	Documento de Testes	97

Lista de Figuras

1.1	Representação do efeito de bola de neve.	4
2.1	Volume de vendas no 1º trimestre de 2013 e percentagem da cota de mercado em comparação com 2012.[IDC13]	8
2.2	Processo de desenvolvimento com a plataforma PhoneGap.	14
2.3	Processo de comunicação da API do PhoneGap.	15
2.4	Funcionalidades nativas suportadas pela plataforma PhoneGap.[Phob]	16
2.5	Processo de comunicação da aplicação PhoneGap com o repositório de dados. .	16
2.6	Imagens da aplicação Indeed.	20
2.7	Imagens da aplicação Trovit.	21
2.8	Imagens da aplicação Net-emprego.	22
3.1	Imagem ilustrativa do quadro de tarefas da equipa.	26
4.1	Vista global da plataforma e conexões com fontes externas.	35
4.2	Visão birds-eye do JOObian.	38
4.3	Conetores do JOObian.	39
4.4	Componentes do JOObian.	40
4.5	Diagrama físico com especial atenção à escalabilidade.	42
4.6	Diagrama no qual está assente um aplicação híbrida PhoneGap.	43
4.7	Diagrama fluxo de dados no Android entre a WebView e a parte Nativa. . . .	44
4.8	Diagrama fluxo de dados no iOS entre a WebView e a parte Nativa.	45
4.9	Solução da implementação da aplicação móvel JOObian.	46
5.1	Workflow da criação de uma aplicação em PhoneGap.	53
5.2	Diagrama de funcionalidades da aplicação móvel JOObian.	58
5.3	<i>Storyboard</i> de login da aplicação móvel JOObian.	59
5.4	Diagrama de fluxo do ecrã inicial da aplicação móvel JOObian.	60
5.5	Ecrã de “Sign in” da aplicação móvel JOObian.	61
5.6	Ecrã da página de login do LinkedIn.	61
5.7	Diagrama de sequência do protocolo OAuth.	62
5.8	Diagrama de sequência no processo de login pelo LinkedIn.	63
5.9	<i>Storyboard</i> do registo da aplicação móvel JOObian.	65
5.10	Ecrã 2 de “Sign up” da aplicação móvel JOObian.	66
5.11	Diagrama de fluxo no registo de conta na plataforma JOObian.	67
5.12	Ecrã 6 da primeira página da fase de registo.	69
5.13	Ecrã 9, segundo passo da fase de registo.	70

5.14	Diagrama de sequencia do processo de comunicação entre o cliente e o servidor.	71
5.15	Ecrã 12 da terceira fase de registo.	71
5.16	<i>Storyboard</i> da pilha de ofertas de emprego.	73
5.17	Ecrã 3 com o detalhe da ofertas de emprego.	74
5.18	Adicionar ou remover skill do perfil.	75
5.19	<i>Highlight</i> das skills nas ofertas de emprego.	75
5.20	Botão para ver oferta de emprego no provedor da oferta de emprego.	76
5.21	Ecrã da pesquisa com as duas pilhas de ofertas de emprego.	76
5.22	<i>Storyboard</i> dos favoritos do utilizador.	77
5.23	<i>Storyboard</i> do perfil de utilizador.	78
5.24	Exemplo do alerta ao atingir patamar mínimo de cinco skills.	78
5.25	Ecrã de estatísticas do utilizador sobre as ofertas de emprego recomendadas.	79
5.26	Ecrã de estatísticas do utilizador.	79
5.27	Valores representativos das estatísticas.	80
5.28	<i>Storyboard</i> das definições do utilizador na aplicação móvel JOObian.	81
5.29	Lista de países selecionados pelo utilizador.	81
5.30	Definições: Notificações e uso da Localização.	82
5.31	Notificação enviada pela plataforma JOObian.	82
5.32	Ecrã de alteração de palavra passe.	82
5.33	Ecrã “About us”.	83
6.1	Número de funcionalidades suportadas pelos navegadores dos dispositivos móveis, das 555 funcionalidades do HTML5 em cada versão do sistema operativo ou navegador móvel.[Nie]	90

Lista de Tabelas

2.1	Tabela comparativa das aplicações de ofertas de emprego.	23
3.1	Lista de casos de uso de utilizador.	32
3.2	Lista de casos de uso da aplicação.	32
3.3	Lista de casos de uso da oferta de emprego.	32
3.4	Lista de casos de uso da pesquisa.	33
3.5	Lista de casos de uso da pesquisa.	33
3.6	Lista de casos de uso do perfil.	33
3.7	Lista de casos de uso da interface externa.	33
3.8	Lista de casos de uso de performance.	34
3.9	Lista de casos de uso do sistema.	34
5.1	Lista de serviços do suporte.	51
5.2	Lista de serviços da autenticação.	51
5.3	Lista de serviços do utilizador.	52
5.4	Lista de serviços da oferta de emprego.	52

Acrónimos

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
BCL	Base Class Library
CSS	Cascading Style Sheets
CSS3	Cascading Style Sheets, versão 3
CLI	Command-Line Interface
DOM	Document Object Model
EAP	Enterprise Application Platform
EAR	Enterprise ARchive
ECM	Enterprise Content Management
EJB	Enterprise JavaBeans
EKM	Enterprise Knowledge Management
GPS	Global Positioning System
HTML	Hypertext Markup Language
HTML5	Hypertext Markup Language, versão 5
HTTP	Hypertext Transfer Protocol
I/O	Input/Output
IDE	Integrated Development Environment
IP	Internet Protocol
IIS	Internet Information Server
Java EE	Java Platform, Enterprise Edition
JSON	JavaScript Object Notation
MD5	Message-Digest Algorithm 5

MVC	Model-view-controller
OWL	Web Ontology Language
PIB	Produto Interno Bruto
RAM	Random-access memory
REST	Representational State Transfer
RDFS	Resource Description Framework Schema
RSS	Rich Site Summary
SDK	Software Development Kit
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SPA	Single Page Application
SSL	Secure Sockets Layer
UI	User Interface
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XHR	XML Http Request
XML	Extensible Markup Language

Capítulo 1

Introdução

“Escolhe um trabalho de que gostes, e não terás que trabalhar nem um dia na tua vida.” - Confúcio

O mercado de trabalho em Portugal caracterizou-se por uma reduzida taxa de desemprego e elevada taxa de emprego nas décadas de 80 e 90 alterando-se significativamente nos anos seguintes. Neste momento os portugueses enfrentam o dilema de ficar no seu país e encarar o desemprego ou deixar-se levar pela corrente migratória e partir na busca de novas oportunidades noutros países.

Com o crescimento da taxa de alfabetismo e a formação académica dos portugueses, estes têm cada vez mais habilitações e conhecimentos em determinadas áreas de estudo. Devido à situação económica que o país atravessa neste momento, surgiu a ideia de criar uma plataforma que fosse capaz de ajudar as pessoas que se encontram desempregadas ou à procura de melhorar a sua situação profissional, de modo a encontrar um emprego que se enquadre nas suas qualificações académicas, quer em Portugal quer noutros países.

1.1 Estrutura da Dissertação

O projeto foi apresentado pela Maisis - Information Systems, Lda. e realizado nas instalações da mesma no contexto da disciplina de Dissertação do Mestrado Integrado em Engenharia de Computadores e Telemática da Universidade de Aveiro. O objetivo deste projeto consiste na criação de uma aplicação móvel que permita simplificar o processo de procura de emprego e a gestão de carreira do utilizador.

Este documento visa expor o trabalho que foi feito durante o período de realização da dissertação. O mesmo encontra-se dividido em 8 capítulos dos quais em seguida será feita uma breve descrição para permitir ao leitor ter uma ideia do que cada capítulo trata.

No presente capítulo é feita uma contextualização do projeto. Esta consiste numa breve descrição da empresa responsável pelo mesmo e da exposição dos objetivos principais do projeto.

Após o capítulo introdutório, são abordados no capítulo 2 (Estado da Arte) vários temas que estão relacionados com o desenvolvimento e modelação de aplicações para *smartphones*, contextualizando assim o projeto e quais os passos a dar nas fases seguintes.

No capítulo 3 (Planeamento e Metodologia), apresentamos a metodologia utilizada e também é apresentado o planeamento inicial do projeto, bem como as alterações que o mesmo tenha sofrido face aos constrangimentos que foram surgindo.

Os três capítulos seguintes (Definição de requisitos, Arquitetura e desenho e Desenvolvimento e Integração) são dedicados exclusivamente ao projeto que motivou o estudo nesta área. No capítulo 4 (Definição de requisitos), é descrito o processo de avaliação dos requisitos do sistema e são apresentados as pessoas e atores que foram considerados para o mesmo. No capítulo 5 (Arquitetura e desenho), é apresentado o modelo da arquitetura que dá suporte à aplicação móvel e é também mostrada a arquitetura de uma aplicação desenvolvida em PhoneGap e todos os mecanismos necessários para a comunicação entre a parte web e nativa. No capítulo 6 (Desenvolvimento e Integração), é descrito todo o trabalho que foi feito durante o desenvolvimento e todas as considerações que foram tidas durante este processo.

No capítulo 7 são descritos os vários tipos de testes realizados à aplicação móvel JOObian.

Finalmente, no capítulo 8, são expostas as conclusões. Aqui são apresentados os resultados e é feita uma breve reflexão sobre o trabalho realizado bem como as projeções futuras que são propostas para a continuação e desenvolvimento do projeto.

1.2 A Empresa

A Maisis - Information System, Lda. é uma empresa portuguesa ligada ao desenvolvimento de software e tem no momento cerca de 90 funcionários. Foi fundada em 1994 e tinha na altura como principal atividade o mercado das telecomunicações. Ao longo dos anos, a empresa foi adquirindo um vasto conhecimento e experiência nas últimas tecnologias da indústria de desenvolvimento de software. A partir de 2004, a sua atividade passou a ser direccionada ao desenvolvimento de novos produtos ligados à gestão de conteúdo empresarial (ECM) e à gestão de conhecimento empresarial (EKM).

A Maisis - Information System, Lda. tem feito um grande investimento em pesquisa e desenvolvimento, o que lhe tem permitido a participação em projetos internacionais e a construção de várias parcerias estratégicas, tendo assim a oportunidade de partilhar o seu *know-how* com companhias de todo o mundo, fazendo com que se tornem mais competitivas e avançadas tecnologicamente.

O primeiro contacto com a Maisis - Information System, Lda. foi tido durante a escolha da dissertação e o contacto com as instalações da empresa e equipa de trabalho da mesma influenciou de forma positiva a decisão tomada. A escolha desta dissertação prendeu-se com o interesse do estagiário pelo campo das aplicações móveis e com a intenção de fornecer algo útil para a sociedade. Para além deste fator, o estagiário viu esta oportunidade para ter a sua primeira experiência com o mercado de trabalho na sua área.

1.3 Descrição do Sistema

A aplicação móvel Joobian pretende efetuar o cruzamento das ofertas de emprego existentes no mercado com o perfil de cada utilizador, possibilitando dessa forma que este último obtenha as ofertas de emprego que melhor se adequam ao seu perfil.

O cruzamento efetuado tem como parâmetros os diversos pontos do perfil do utilizador e as diferentes características da oferta. Inicialmente tudo pode ser efetuado de uma forma semiautomática, ou seja, com a interligação ao LinkedIn onde o utilizador já tem definido o

seu perfil e com análise ao texto da oferta esse cruzamento torna-se possível. No entanto para que a garantia de bons resultados seja cada vez maior o utilizador poderá alterar parâmetros no seu perfil tentando que as ofertas sejam mais adequadas ao mesmo.

O que se pretende como produto final é uma aplicação móvel que cumpra com os requisitos funcionais, com as boas práticas de desenvolvimento não só ao nível do código, mas também em relação à modelação da interface.

1.4 Motivação e Objetivos

A plataforma de emprego JOObian será construída com o intuito de ter o maior número de dados de perfis de candidatos como de ofertas de emprego, pois quantos mais dados houverem na plataforma, mais valor será possível gerar através desta. Sendo este o principal objetivo, é necessário que a aplicação móvel aqui detalhada maximize esta estratégia e para isso é vital cumprir com os seguintes pontos-chave:

- Facilitar e incentivar que mais pessoas se juntem à plataforma e registem os seus dados – Os utilizadores ao fazerem uso da aplicação e a definirem o seu perfil dotam a plataforma de nova informação que contribui positivamente para que esta possa dar resultados mais assertivos a utilizadores com diferentes perfis.
- Possibilidade de integração com sistemas já existentes – Havendo integração com outros sistemas, é possível atrair também empresas que já utilizem outros sistemas para este efeito.
- Utilização de fontes externas que permitam obter dados de perfis e ofertas de emprego – Isto permite a existência prévia de alguns dados no sistema quando for inicializado, não havendo dependência de adição de dados diretamente por utilizadores, e uma maior variedade destes devido à obtenção de várias fontes (e.g. dados extraídos através do LinkedIn ou CV Europass).

1.5 Público Alvo

A plataforma JOObian tem como principais utilizadores as seguintes entidades:

- Candidatos – Pessoas que estejam interessadas em consultar informação sobre as ofertas de emprego, sendo as próprias pessoas a principal fonte de perfis de candidatos.
- Empresas – Empresas que tencionam encontrar candidatos para as ofertas de emprego que possuem, e das quais serão a principal fonte.

Note-se que este projeto irá focar apenas o plano dos Candidatos. Tendo em conta que tanto Candidatos como Empresas adicionam ao sistema os dados que interessam um ao outro, há aqui a criação de um efeito de bola de neve que vai assegurar o incremento do valor da plataforma ao longo do tempo.

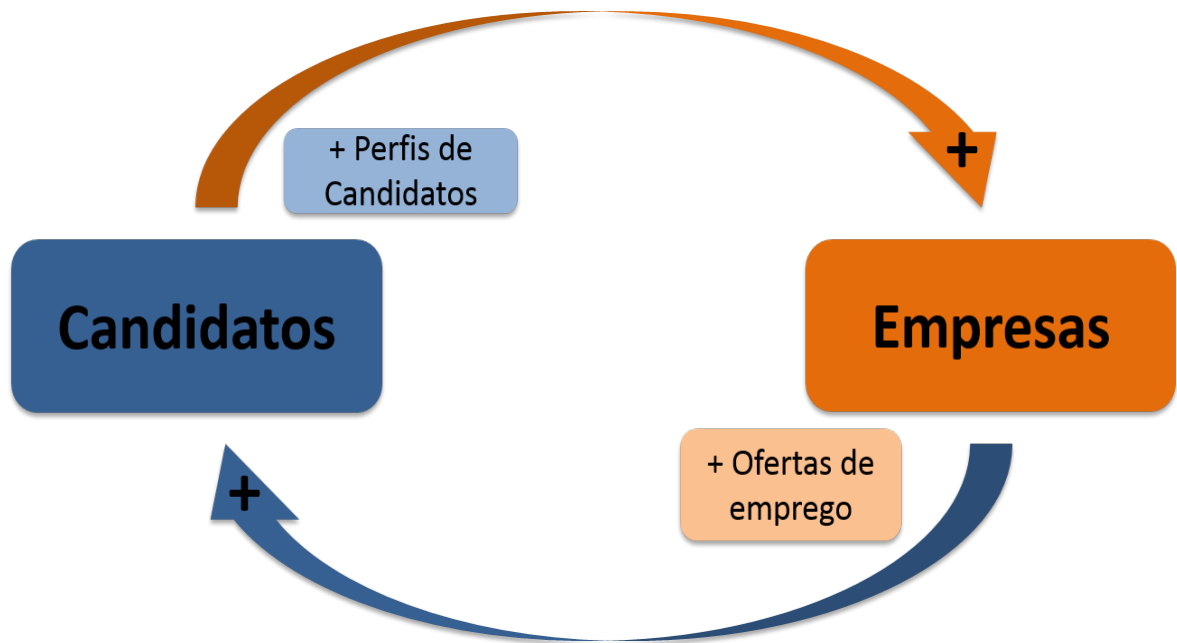


Figura 1.1: Representação do efeito de bola de neve.

Numa segunda fase, tendo os dados suficientes no sistema, será possível fazer uma análise que permita oferecer aos seus utilizadores um serviço mais direcionado às suas necessidades através de recomendações e métricas obtidas dessa análise. Isto irá permitir uma correspondência entre candidatos e ofertas de emprego.

1.6 Funcionalidades Gerais

A aplicação a desenvolver será baseada numa infra-estrutura de serviços a fornecer pela Maisis - Information Systems, Lda, que suportará toda a lógica de negócio e de acesso aos dados necessários para a concretização das funcionalidades requeridas. Portanto, o desenvolvimento da aplicação envolverá essencialmente o aspeto visual e a camada de acesso aos serviços.

A aplicação deverá conter as seguintes funcionalidades:

- Registo de novos utilizadores via LinkedIn ou registo interno.
- Autenticação no sistema.
- Ver as ofertas de emprego recomendadas para o perfil do utilizador.
- Ordenar as ofertas de emprego por relevância, proximidade, data, salário ou por poupança salarial.
- Permitir procurar ofertas de emprego por título, habilitações, empresa ou localização.
- Permitir adicionar e remover ofertas de emprego aos favoritos.

- Ver e editar o perfil do utilizador.
- Apresentação estatística do perfil de utilizador sobre as ofertas emprego recomendadas para ele.
- Ver informação do custo de vida da cidade da oferta de emprego.
- Ver e editar as definições que o utilizador tem na aplicação, como países, notificações e geolocalização.

Para além destas funcionalidades a aplicação tem que proporcionar uma boa experiência de utilização.

Capítulo 2

Estado da Arte

Em todo o mundo, tem-se assistido ao forte crescimento dos dispositivos móveis e o caminho que estão a tomar para se tornarem um dos principais instrumentos na utilização do dia a dia. Começa-se a notar que as pessoas estão a deixar de lado os seus computadores e notebooks devido à maior praticidade oferecida pelos dispositivos móveis. Devido a este crescimento o desenvolvimento de aplicações móveis tornou-se uma área de grande relevo junto da comunidade de *Developers*, bem como das empresas que têm interesse neste mercado em expansão, afinal a maior parte do tempo gasto nos dispositivos móveis é na utilização das aplicações que estes oferecem.

Atualmente temos alguns sistemas operativos móveis em forte concorrência no mercado global. Temos como principais rivais desta lista o Android e o iOS, número um e dois do ranking mundial da lista de sistemas operativos para dispositivos móveis, que combinados representam 92,3% de todos os smartphones vendidos no mundo, com o Windows Phone a passar o BlackBerry para um seguro 3º lugar.

A Figura 2.1 é bem ilustrativa do poder que o Android tem em relação aos outros sistemas. Existem dois fatores que contribuem para este crescimento: o primeiro deve-se ao facto de o Android ser um sistema operativo móvel open source; o segundo fator, em consequência do primeiro, está relacionado com a grande segmentação de dispositivos com este sistema operativo.

Até o Android ganhar relevo no mercado, o iOS era o sistema operativo que figurava o topo do ranking dos sistemas operativos móveis. Como este sistema só corre nos equipamentos da Apple, não existe segmentação e só disponibilizam um conjunto mínimo de equipamentos (iPhone, iPad, iPod). Outro fator para a perda do 1º lugar deve-se ao facto destes equipamentos terem um custo elevado para a maioria dos consumidores.

O Windows Phone é a evolução do Windows Mobile da Microsoft e o número de utilizadores tem crescido ano após ano com grandes campanhas de marketing e pela compra da Nokia por parte da Microsoft. Como é recente ainda poucas empresas estão a produzir equipamentos com este sistema, mas tem do seu lado a possibilidade de segmentação e de haver equipamentos de baixa gama, aumentando assim o número de consumidores.

Foram também aparecendo outros sistemas com grande potencial de mercado como o BlackBerry 10, Ubuntu, FirefoxOS, Tizen, entre outros, que vão sendo adaptados pelas empresas. O potencial de crescimento dos diferentes sistemas passa pelo leque de aplicações que a respetiva loja oferece. Se não houver programadores a desenvolver aplicações para os

sistemas operativos, isso torna-se a principal barreira para ter sucesso junto dos consumidores.

Operating System	1Q13 Shipment Volume	1Q13 Market Share	1Q12 Shipment Volume	1Q12 Market Share	Year over Year Change
Android	162.1	75.0%	90.3	59.1%	79.5%
iOS	37.4	17.3%	35.1	23.0%	6.6%
Windows Phone	7.0	3.2%	3.0	2.0%	133.3%
BlackBerry OS	6.3	2.9%	9.7	6.4%	-35.1%
Linux	2.1	1.0%	3.6	2.4%	-41.7%
Symbian	1.2	0.6%	10.4	6.8%	-88.5%
Others	0.1	0.0%	0.6	0.4%	-83.3%
Total	216.2	100.0%	152.7	100.0%	41.6%

Figura 2.1: Volume de vendas no 1º trimestre de 2013 e percentagem da cota de mercado em comparação com 2012.[IDC13]

Neste capítulo vão ser abordados os três tipos de desenvolvimento de aplicações móveis e quais os aspetos que as diferenciam. Vai ser dada uma visão de quais são as plataformas Cross-platform que existem e quais as *frameworks* mais usadas no desenvolvimento de aplicações web ou híbridas. É feita ainda uma análise das aplicações móveis de ofertas de emprego mais relevantes que existem na atualidade.

2.1 Nativo, Web App ou Híbrido

Quando se fala em aplicações móveis é comum ouvir-se termos como aplicações nativas ou aplicações web ou até mesmo aplicações híbridas. Impõem-se portanto saber qual delas é a melhor estratégia para desenvolver uma aplicação, tendo em conta todos os requisitos e o público alvo. Se por um lado temos a abordagem mais sólida que é desenvolver aplicações que corram nativamente no sistema operativo dos dispositivos, por outro temos as aplicações web mais versáteis que são baseadas em HTML5, JavaScript e CSS3.

Mas na realidade qual é a diferença entre elas?

Aplicações Nativas:

As aplicações nativas são desenvolvidas especificamente numa plataforma, no caso do iOS todo o desenvolvimento é feito em Objective-C, para Android usamos Java e Windows Phone utilizamos C#. Desta forma as aplicações tiram o máximo partido de todos os recursos do dispositivo - que podem ser a câmara, o GPS, o acelerómetro, a bússula, o correio eletrónico, a lista de contactos, entre todos os outros recursos existentes bem como a instanciação de outras aplicações. As aplicações nativas podem usar o sistema de notificações do dispositivo e tem a possibilidade de poder trabalhar offline. As aplicações desenvolvidas podem ser facilmente encontradas e instaladas a partir das lojas como o Google Play¹, iTunes App Store² ou Windows Phone Store³.

Aplicações Web:

As aplicações web não são aplicações reais. Na realidade são um website que de muitas maneiras se fazem parecer com uma aplicação nativa. Estas aplicações são executadas num navegador e normalmente implementadas em HTML5. Para aceder a estas aplicações primeiro é necessário abrir um navegador e aceder a um URL que contenha um site desenvolvido com o propósito de se parecer com uma aplicação nativa. Há medida que mais sites usam HTML5 a distinção entre páginas web e aplicações nativas tornam-se difíceis de identificar.

Aplicações Híbridas:

As aplicações híbridas conjugam a perspetiva web e a perspetiva nativa, isto é, o core em que estão assentes é nativo mas o desenvolvimento é feito em HTML5, JavaScript e CSS3. Como nas aplicações nativas estas são instaladas a partir de lojas e tiram partido dos recursos do dispositivo. A aplicação é desenvolvida sobre HTML e é processada por um navegador que está dentro da aplicação. Muitas vezes estas aplicações não passam de um reencaminhamento para uma aplicação web, desta forma garantem que a aplicação está presente nas lojas e não foi gasto um esforço significativo no desenvolvimento de uma aplicação para as mais variadas plataformas. Estas aplicações são muito populares porque permitem desenvolvimento cross-platform, ou seja, os mesmo componentes de código HTML podem ser reutilizados em diferentes sistemas móveis, reduzindo significativamente os custos de desenvolvimento. Existem diversas ferramentas como o PhoneGap, Sencha Touch, entre outras, onde as aplicações são desenvolvidas como se trata-se de uma aplicação web usando HTML5, JavaScript e CSS3, e que depois são “renderizados” num navegador embutido no SDK padrão da plataforma nativa e o resultado final é uma aplicação nativa. Pode-se dizer que atuam como um *wrapper*

¹<https://play.google.com/store>

²<https://itunes.apple.com/pt/genre/ios/id36?mt=8>

³<http://www.windowsphone.com/pt-pt/store>

de uma página web e que disponibiliza o acesso aos recursos do dispositivo através da API's.

Cada um destes tipos de aplicações/abordagens tem as suas vantagens e desvantagens, como por exemplo:

Recursos do dispositivo: As aplicações nativas são desenvolvidas especificamente para explorar na total plenitude os recursos e UI dos dispositivos móveis. Por exemplo o uso da máquina fotográfica, o GPS, gestos, notificações, entre outros. As aplicações híbridas apesar de serem desenvolvidas em contexto web, também têm acesso a alguns destes recursos. Por outro lado as aplicações web não podem explorar plenamente os recursos específicos que determinado dispositivo oferece, por estarem a correr dentro de um navegador.

Funcionamento Offline: Uma aplicação nativa é melhor quando pretendemos que esta trabalhe em modo offline. Nas aplicações web temos a cache do navegador, mas esta é limitada em relação à parte nativa.

Visibilidade: O conteúdo é muito mais visível na web do que numa aplicação. Quando uma pessoa necessita de obter informação de algo, por norma acede a um motor de pesquisa num computador, não vai à loja do sistema operativo móvel procurar uma aplicação e depois obter a resposta dentro desta, isto deve-se ao facto do dispositivo ter um ecrã de dimensões reduzidas e não ser tão prático como um computador.

Desempenho: Neste campo as aplicações nativas ganham por completo as aplicações web. Este é um aspeto crítico e tem um papel fundamental na experiência de utilização de qualquer aplicação. Uma aplicação pouco otimizada na execução de certas funções ou com muito latência na sua interação pode ser à partida excluída pelo simples facto de não proporcionar uma boa experiência de utilização. O desempenho tem que ser visto como um campo subjetivo, pois existem muitas variáveis que podem interferir no funcionamento de uma aplicação, como por exemplo, o hardware do dispositivo, a memória livre disponível, a estabilidade do sistema, a largura de banda da ligação à Internet, entre outros. As métricas mais comuns para avaliar o desempenho de uma aplicação são, a latência e o tempo de execução que uma função ou operação demora, mas existem outras variáveis para quantificar o desempenho.

A latência é uma variável métrica de muita relevância em aplicações web e para todas as aplicações que recebem ou enviam dados pela Internet. Outro aspeto tem a ver com o tempo que uma aplicação demora a ser carregada em memória e no caso do JavaScript há uma latência associada à interpretação do código.

Como as aplicações nativas já estão presentes no dispositivo não é necessário transferi-las, o que já não é verdade nas aplicações web, este fator causa uma menor latência na execução da aplicação. Por outro lado temos a fluidez que uma interface nativa proporciona em comparação a uma híbrida, sendo que esta última está dependente do navegador e do motor de Webview do sistema. Devido a este fator as aplicações híbridas têm um tempo de execução maior, não só porque estão a correr por cima de outra aplicação como têm que fazer *parsing* do JavaScript. As aplicações nativas tem a vantagem de ser desenvolvidas, compiladas e otimizadas para o sistema operativo móvel em que funcionam.

Instalação: Nos últimos tempos tem-se assistido a um grande desenvolvimento nos prin-

cipais componentes de hardware dos dispositivos móveis, como o processador, a placa gráfica e memória RAM. Devido a estes fatores as aplicações começam a tornar-se mais pesadas e exigentes em termos de recursos, e os utilizadores com dispositivos mais limitados têm de ponderar para efetuar a instalação de uma aplicação sendo que essa instalação seja justificada, e é aqui que as aplicações web ganham vantagem contra as nativas ou híbridas, pois estas correm no navegador do dispositivo.

Estar presente na loja torna todo o processo de instalação mais fácil e confiável, porque são aplicações que estão assinadas e que passaram por um processo de aprovação. Ao instalar uma aplicação é criado um ícone no sistema e torna o acesso a estas mais fácil e rápido, em vez de estar a abrir um navegador, aceder a um link e depois guardar um marcador. Em termos estatísticos é possível verificar que os utilizadores não tem o hábito de guardar marcadores no navegador quando usam dispositivos móveis.

Manutenção: A manutenção de uma aplicação nativa para um *developer* torna-se mais difícil se houver várias versões da aplicação em diferentes plataformas. As alterações têm que ser compiladas e enviadas de novo para as lojas. Por outro lado, a manutenção de uma aplicação web ou híbrida é tão simples como a manutenção de uma página web, e isso pode ser feito com a frequência ou tão frequentemente quanto necessário. As aplicações híbridas apesar da fácil manutenção, também têm que passar pelo processo de compilação e ser enviadas de novo para as lojas.

Independência da plataforma: Como todas as versões dos dispositivos móveis suportam HTML, desenvolver uma aplicação web ou híbrida torna o processo de desenvolvimento mais fácil, pois é compatível com todas as versões e plataformas. Já nas versões nativas desenvolver para todas as versões torna o processo mais difícil porque as novas versões trazem *features* que as versões antigas não disponibilizam.

Restrições de conteúdo, processo de aprovação e taxas: Lidar com terceiros que impõem regras sobre o conteúdo e design pode ser cansativo, tanto em termos de tempo como dinheiro. As aplicações nativas e híbridas tem que passar por um processo de aprovação e restrições de conteúdo por parte das lojas, enquanto que a web é livre para todos. Além deste fator é necessário ainda contar com a percentagem no valor de vendas das aplicações que reverte a favor das lojas, o que leva a uma diminuição dos lucros por parte dos *developers* ou empresas.

Custo de desenvolvimento: As aplicações desenvolvidas com ferramentas cross platform ou web, geralmente diminuem consideravelmente o custo de um projeto, principalmente quando várias plataformas estão envolvidas. Na abordagem do desenvolvimento nativo este pode ter um alto custo, isto deve-se ao facto de termos que construir n versões da mesma aplicação para as várias versões e plataformas existentes. Geralmente isto multiplica o custo do projeto e o valor cobrado pela aplicação nas lojas. A este custo podemos adicionar a manutenção, como existem várias bases de código, cada uma delas pode conter erros que não se encontram nas outras. Outro custo é a obtenção de uma equipa que consiga desenvolver para múltiplas plataformas ou a obtenção de uma equipa para cada plataforma.

Interface: Não menos importante que as outras, pois sem uma boa interface os utilizadores não ficam na aplicação muito tempo. Para manter uma consistência com o sistema móvel

e com a maioria das outras aplicações o caminho a seguir é a implementação em nativo. Com isto não quer dizer que não se possa fornecer uma boa experiência com aplicações híbridas ou web, apenas significa que a interface não possa estar de acordo com aquilo a que estão habituados.[Ral13]

Conclusão: As principais formas de desenvolvimento de aplicações móveis são as aplicações nativas, as aplicações híbridas e as aplicações web. Não há uma solução única, cada uma tem os seus pontos fortes e fracos. A escolha de um método contra o outro depende da necessidade específica de cada *developer* ou empresa. Isto é, se for necessário o uso de algum recurso mais complexo de hardware ou se desejar um alto desempenho gráfico, então uma aplicação nativa é o mais indicado. As aplicações web ou híbridas são indicadas para quem quer disponibilizar uma aplicação nas diversas plataformas e não tem muito tempo ou dinheiro para investir no desenvolvimento de uma aplicação para cada plataforma.

2.2 Cross-Platform - Ferramentas de desenvolvimento

“In computing, cross-platform, or multi-platform, is an attribute conferred to computer software or computing methods and concepts that are implemented and inter-operate on multiple computer platforms.”[Sta]

Quando se considera desenvolver uma aplicação iOS ou Android, a ideia principal é desenvolver em Objective-C ou Java, respetivamente. Contudo, com o passar dos anos, um novo ecossistema de plataformas para desenvolver aplicações móveis tem imergido para ajudar os programadores e empresas a desenvolver as suas aplicações, poupando tempo e dinheiro.

Nesta secção vamos explorar algumas das ferramentas mais utilizadas para o desenvolvimento de aplicações cross-platform.

2.2.1 Xamarin

Xamarin é a única plataforma no seu meio a utilizar uma só linguagem de programação o C#, uma biblioteca de classes e um motor que funciona com as três principais plataformas iOS, Android e Windows Phone, e ainda a compilação nativa (não são aplicações interpretadas) que mesmo para jogos mais exigentes tem a performance pretendida.

O Xamarin é a única plataforma que combina o poder das linguagens nativas e acrescenta uma série de poderosos recursos que incluem:

- Completa ligação com os SDKs nativos — O Xamarin contém ligações para quase todos os elementos dos SDK tanto de iOS como de Android. Permitindo assim a verificação dos tipos em compilação e durante o desenvolvimento, Isto leva a menos erros de execução e aplicações de maior qualidade.
- Interoperabilidade entre linguagens de programação — O Xamarin permite chamar diretamente bibliotecas escritas em Objective-C, Java, C ou C++, permitindo assim usar uma grande variedade de código que já foi criado por outras pessoas. Permite ainda facilmente ligar bibliotecas nativas Objective-C e Java usando uma sintaxe declarativa.

- Construção de uma linguagem moderna — As aplicações em Xamarin são escritas em C#, uma linguagem moderna que inclui melhorias significativas sobre Objective-C e Java, tais como Dynamic Language Featured, Functional Constructs como *Lambdas*, *LINQ*, *Parallel Programmin*, entre outras.
- Ampla biblioteca de classes — As aplicações em Xamarin usam .NET BCL, uma enorme coleção de classes que têm recursos abrangentes e ágeis tais como XML, Base de Dados, Serialização, I/O, String e suporte para rede. Além disso, o código C# existente pode ser compilado para ser usado nas aplicações, dando acesso a milhares de bibliotecas que não estão abrangidas no BCL.
- Suporte para Mobile Cross Platform — O Xamarin oferece suporte cross-platform para as três principais plataformas móveis iOS, Android e Windows Phone. As aplicações podem ser escritas para partilhar até 90% do seu código, e as bibliotecas do Xamarin oferecem a API para aceder aos recursos que as três plataformas tem em comum. Isto pode reduzir significativamente os custos de desenvolvimento e de tempo para os programadores que querem desenvolver para as três plataformas móveis mais populares.

Por causa do abrangente conjunto de recursos do Xamarin, ele preenche uma lacuna para os programadores que querem usar uma plataforma moderna para desenvolver as suas aplicações móveis cross-platform.[Xam14]

2.2.2 MoSync

MoSync é uma ferramenta de desenvolvimento de software SDK grátis e open source para aplicações móveis de crossplatforms e baseada em programação multilingue.[Pri] Este oferece suporte para vários tipos de sistemas operativos móveis, incluindo Windows Phone, Android, Symbian, iOS e BlackBerry.[Cli14] O público alvo do MoSync são os programadores web que querem entrar no espaço móvel, como programadores convencionais com conhecimentos de C/C++. O MoSync gera aplicações nativas usando C/C++, HTML5/JavaScript ou a combinação das duas. A partir deste código base, o MoSync consegue gerar pacotes de aplicações para os diferentes dispositivos móveis, numa gama ampla de sistemas operativos. Uma das vantagens é o suporte para “Native UI” nas plataformas Android e iOS, permitindo criar aplicações com aparência nativa. Foi ainda introduzido no MoSync uma tecnologia designada Wormhole⁴, que faz o *data bind* entre o JavaScript e as APIs subjacentes do MoSync, permitindo ao programador as funcionalidades da WebView.[Unk12]

2.2.3 PhoneGap

PhoneGap é uma plataforma open source para um rápido desenvolvimento de aplicações móveis usando HTML5, JavaScript e CSS3.

Criar aplicações para cada dispositivo como iPhone, Android, Windows Mobile e outros, requer diferentes *frameworks* e linguagens de programação. Neste sentido o PhoneGap resolve este problema usando tecnologias baseadas em web, sendo que o produto final é uma aplicação que pode ser distribuída pelo ecossistema dos diversos dispositivos móveis.

⁴<http://www.mosync.com/docs/sdk/js/guides/wormhole/html5-javascript-wormhole/index.html>

A WebView usada pelo PhoneGap é a mesma WebView usada pelo sistema operativo. No iOS esta é `UIWebView.class`; no Android, é `android.webkit.WebView`. Uma vez que existem diferenças entre os motores de renderização do WebView entre sistemas operativos, é preciso ter em conta estas diferenças na implementação da interface do utilizador.

A Figura 2.2 mostra o conceito geral no processo de desenvolvimento de uma aplicação em PhoneGap.

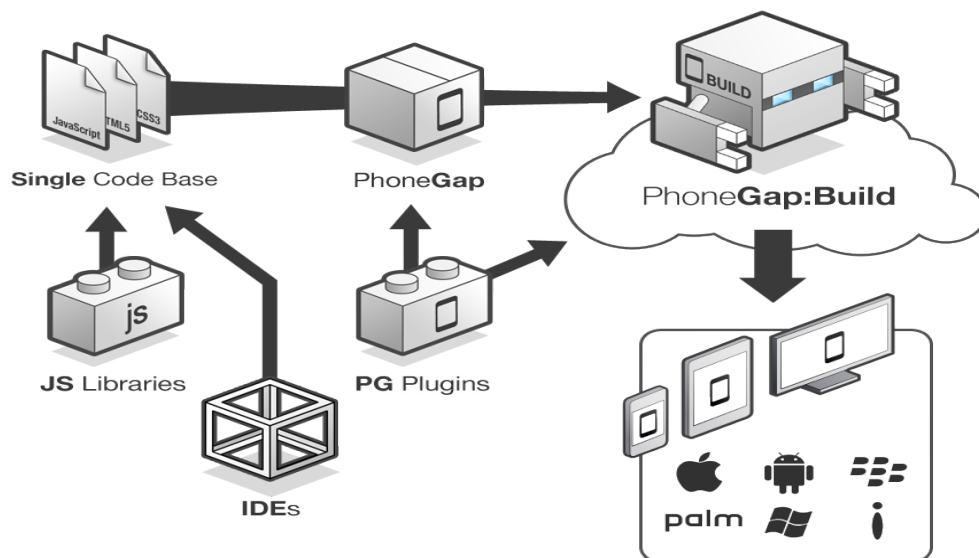


Figura 2.2: Processo de desenvolvimento com a plataforma PhoneGap.

Os principais requisitos identificados para a aplicação JOObian eram, ser mutiplataforma e que estivesse presente nas lojas de aplicações dos sistemas operativos móveis. Nesse sentido e dados os mais baixos custos de desenvolvimento em comparação ao desenvolvimentos nativo e à flexibilidade em gerar uma aplicação para os diferentes sistemas, a opção recaiu sobre o PhoneGap. Esta decisão teve também como fator decisivo a enorme comunidade que contribui ativamente para a adopção da plataforma, disponibilizando diferentes plugins que possibilitam a integração de PhoneGap com diferentes plataformas móveis e com as funcionalidades nativas de cada sistema. Em função desta escolha irá ser detalhado no ponto seguinte com maior detalhe os diferentes aspetos inerentes a esta plataforma.

PhoneGap API

A API PhoneGap fornece uma interface de programação que permite aceder às funcionalidade do sistema operativo como, geolocalização, contactos, sensores e muito mais. Toda a lógica da aplicação é construída usando JavaScript e a API do PhoneGap lida com a comunicação com o dispositivo.

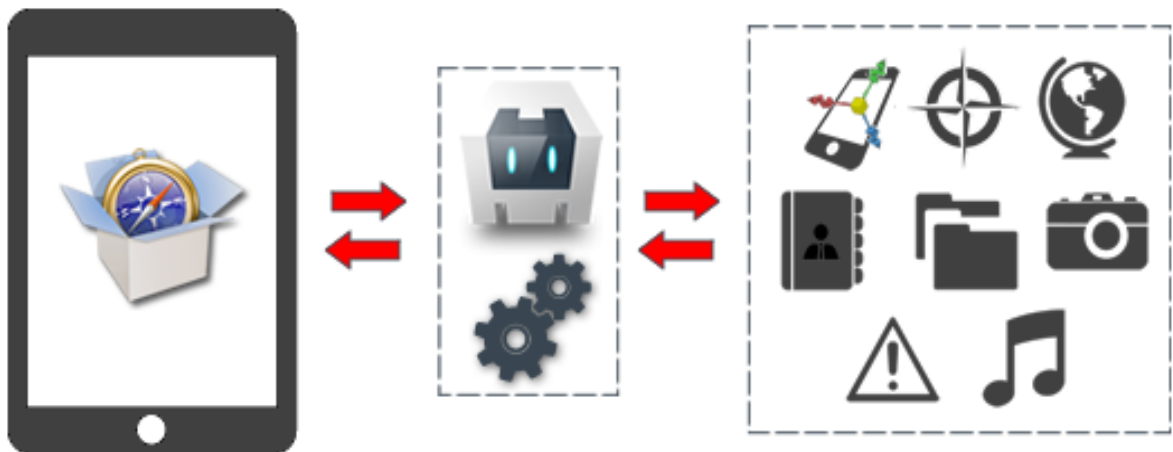


Figura 2.3: Processo de comunicação da API do PhoneGap.

O PhoneGap também fornece um conjunto de mecanismos de comunicação entre o JavaScript e a parte nativa, que permite aos programadores escrever os seus próprios plugins. Estes plugins são classes nativas criadas e personalizadas de modo a que nas interfaces JavaScript se tenha acesso ao recurso ou funcionalidade nativa que por outro meio não se podia usar. Por exemplo, se fosse necessário saber o endereço IP do dispositivo móvel como o JavaScript não suporta essa funcionalidade, seria necessário criar um plugin para ir buscar essa informação à parte nativa e passá-la ao JavaScript.

Na Figura 2.4 podemos ver quais as funcionalidades nativas disponibilizadas na API do PhoneGap e em que sistemas operativos ou plataformas eles são suportados. Pode-se observar que nas três principais plataformas o acesso às funcionalidades nativas é completo.

	iPhone / iPhone 3G	iPhone 3GS and newer	Android	Blackberry OS 6.0+	Blackberry 10	WebOS	Windows Phone 7 + 8	Symbian	Bada
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	X	✓
Contacts	✓	✓	✓	✓	✓	X	✓	✓	✓
File	✓	✓	✓	✓	✓	X	✓	X	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	X	✓	X	X
Network	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	X	X

Figura 2.4: Funcionalidades nativas suportadas pela plataforma PhoneGap.[Phob]

PhoneGap arquitetura de aplicações de alto nível

As especificações de arquitetura das aplicações vão variar caso a caso, contudo a maioria das aplicações que fazem um uso intensivo de dados têm como base a seguinte arquitetura.

A aplicação atua como um cliente para o utilizador interagir. O cliente comunica com um servidor para receber os dados. O servidor lida com a lógica de negócio e comunica com um repositório de dados back-end.

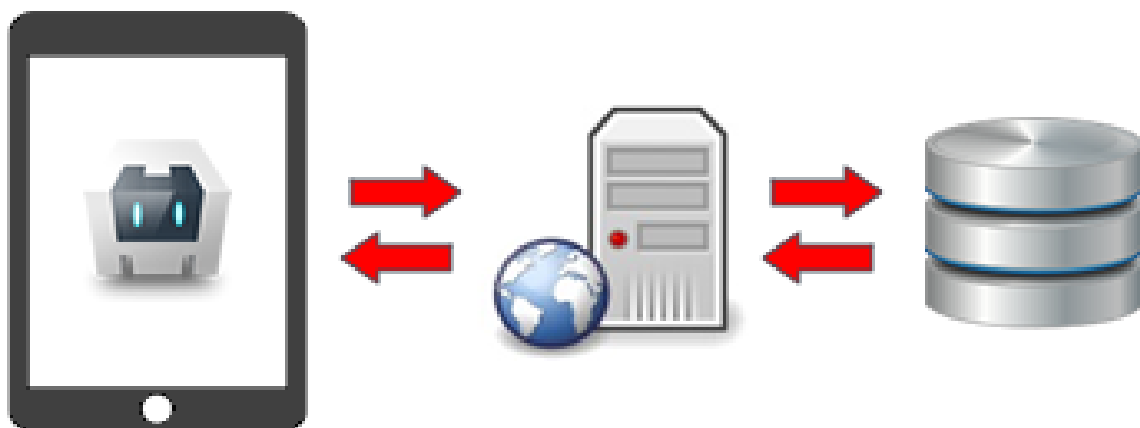


Figura 2.5: Processo de comunicação da aplicação PhoneGap com o repositório de dados.

O servidor normalmente é um servidor web (Apache⁵, IIS⁶, etc...) e usa uma linguagem de programação como por exemplo Java⁷, .NET⁸, PHP⁹, Python¹⁰, CouldFusion¹¹, etc... O PhoneGap é agnóstico à tecnologia de back-end e pode funcionar com qualquer servidor utilizando protocolos padrão da web. O servidor executa a lógica de negócio e os cálculos, lê ou persiste os dados num repósitorio de dados em separado, isto é, normalmente numa base dados relacional.

As aplicações geralmente não comunicam diretamente com as base de dados, a comunicação é encaminhada através de um servidor. Estas são baseada em solicitações HTTP para conteúdo HTML, em serviços REST-ful XML, serviços JSON , ou SOAP (ou websoc-kets se o sistema operativo suportar).

A arquitetura do lado do cliente geralmente usa o modelo de SPA, onde a lógica da aplicação está dentro de uma única página HTML. Todos os dados serão exibidos, atualizando o HTML DOM, os dados são pedidos do servidor utilizando técnicas AJAX, e as variáveis são mantidas na memória dentro do JavaScript.

Arquiteturas multi-página do lado do cliente são suportadas, mas não são recomendados porque perdem-se variáveis em memória ao carregar uma nova página.[And12]

2.3 *Frameworks* de desenvolvimento de aplicações móveis

“A framework is a set of cooperating classes that make up a reusable design for a specific class of software. A framework provides architectural guidance by partitioning the design into abstract classes and defining their responsibilities and collaborations. A developer customizes the framework to a particular application by subclassing and composing instances of framework classes.” [Eri95]

Nesta secção vai-se explorar algumas das *frameworks* mais utilizadas para o desenvolvimento de aplicações híbridas para dispositivos móveis. Devido à explosão dos smartphones no mercado e a necessidade de apoiar todas as plataformas as arquiteturas Cross Platform são a solução mais económica e rápida no desenvolvimento das mesmas. Estas *frameworks* não são mais que bibliotecas ou APIs JavaScript que agilizam o processo de desenvolvimento e que abstraem um conjunto de funcionalidades que são muitas vezes usadas, potenciando assim todo o processo de desenvolvimento. Estas *frameworks* incidem principalmente na interface da aplicação para que o utilizador tenha uma experiência de utilização semelhante às aplicações nativas, tendo assim as vantagens inerentes a esse tipo de interfaces.

2.3.1 jQuery Mobile

jQuery Mobile é uma *framework* móvel cross-platform desenhada para simplificar e aumentar o desenvolvimento de aplicações móveis web ou híbridas integrando HTML5, CSS3,

⁵<http://www.apache.org>

⁶<http://www.iis.net>

⁷<https://www.java.com>

⁸<http://www.microsoft.com/net>

⁹<http://www.php.net>

¹⁰<https://www.python.org>

¹¹<http://www.adobe.com/pt/products/coldfusion-family.html>

jQuery e jQuery UI numa só *framework*. Isto não só é robusto, como sustentável e organizável.[Mar]

jQuery Mobile foi construído sobre o core do jQuery, sendo a curva de aprendizagem reduzida quando se detém conhecimentos sobre a *framework* jQuery. Esta *framework* é compatível com a maioria dos sistemas operativos móveis como por exemplo iOS, Android, Blackberry, Palm WebOS, Symbian, Windows Phone, Opera Mobile, Firefox Mobile e todos os navegadores modernos.[Bra11]

2.3.2 Sencha Touch

Sencha Touch é uma *framework* MVC JavaScript, especialmente concebida para aplicações web móveis para dispositivos touchscreen. Sencha Touch permite aos programadores desenvolverem aplicações web móveis que no final tem um aspeto de aplicação nativa.

O Sencha Touch inclui um conjunto de controlos gráficos de interfaces de utilizador (ou componentes) para uso em aplicações web móveis. Estes componentes estão otimizados para entradas ao toque. Estes componentes são: botões com temas e efeitos do sistema nativo; elementos de formulário como campos de texto para endereços de correio eletrónico, “date picker”, e moradas; “sliders”, “selectors”, e “combo-boxes”; entre outros.

Todos estes componentes podem ser temáticos de acordo com o dispositivo destino. Isto é feito usando Sass¹², uma linguagem de estilos contruído sobre CSS.[Adr13]

2.3.3 AngularJS

AngularJS é uma *framework* JavaScript MVC para desenvolvimento de aplicações web dinâmicas. Ou seja, permite usar HTML como linguagem modelo e permite estender a sintaxe do HTML para expressar os componentes da aplicação de forma clara e sucinta. Algumas das suas principais características são o SPA, Cliente-Side Templates, MVC, Data Bind, Injeção de Dependências, assim como a utilização de Diretivas.[Bra13]

Em seguida serão abordadas as principais características do AngularJS:

SPA - Single Page Application

O AngularJS utiliza o conceito de SPA, assim que a página é carregada, todas as atualizações são realizadas através de pedidos AJAX e de renderizações parciais na página.

Client-Side Templates

O AngularJS renderiza as páginas do lado do cliente, ou seja, no próprio navegador do utilizador. Enquanto que a maioria das aplicações web processam a informação e geram o HTML a ser enviado para o cliente diretamente no servidor, o AngularJS recebe os dados normalmente em formato JSON e processa os dados no navegador, renderizando a página em tempo real diretamente no cliente.

A vantagem deste método é a de não sobrecarregar o servidor uma vez que este só precisa de enviar os dados para o cliente, que ao recebe-os será o responsável pelo processamento e renderização da página. Deixar este processo juntamente com o conceito de SPA, faz com que a performance do aplicação seja superior e que a quantidade de dados trocados entre cliente

¹²<http://sass-lang.com>

e servidor seja menor.

MVC - Model View Controller

O AngularJS utiliza as melhores práticas criadas até hoje para melhorar o seu desempenho, assim como estrutura base ele utiliza o padrão MVC, onde a ideia básica é separar o código entre gestão de dados (model), lógica da aplicação (controller) e apresentação dos dados (view).

No AngularJS a view é o próprio DOM, os controllers são classes JavaScript e o model os próprios dados armazenados em formato de objetos.

Data Bind

O AngularJS para facilitar o desenvolvimento da aplicação utiliza como recurso o Data binding, pois é ele que manipula e trata de atualizar a informação da página.

Através da adição dos controlos diretamente no HTML da página, o AngularJS consegue identificar e mapear os controlos a ser atualizados e trata de tudo automaticamente, não sendo necessário a criação de código extra para nada, só necessita dos controlos, mantendo assim o HTML simples e limpo, de fácil manutenção e visualização por parte do programador e designer.

Injeção de Dependências

O AngularJS usa a injeção de dependências para fornecer recursos extra necessários a aplicação de forma transparente ao utilizador.

Diretivas

Uma das melhores partes do AngularJS é permitir a criação de templates HTML. Através das diretivas é possível criar conteúdo específico que é aplicado a um componente HTML em forma de atributo e irá gerar uma visualização personalizada.[M13]

2.4 Aplicações Móveis de Ofertas de Emprego

Esta secção tem como objetivo explorar algumas aplicações de ofertas de emprego que se podem encontrar atualmente nas lojas de aplicações existentes, de forma a conhecermos as funcionalidades que estas aplicações disponibilizam.

Para filtrar uma grande quantidade de aplicações do mesmo género, restringiu-se o estudo a aplicações com um nível de maturidade elevado (mais de 50 mil utilizadores) e que se focassem dentro da área de estudo.

Os detalhes das aplicações que de seguida serão analisadas foram retirados maioritariamente do Google Play¹³, do iTunes App Store¹⁴ e das páginas web que as representam.

2.4.1 Indeed Jobs

O Indeed é o mais completo motor de pesquisa de emprego. Está presente em mais de 50 países e 26 idiomas, abrangendo 94% do PIB mundial. A aplicação é totalmente gratuita e

¹³<https://play.google.com/store>

¹⁴<https://itunes.apple.com/pt/genre/ios/id36?mt=8>

reúne informação de milhares de sites de emprego, jornais, empresas e agências de recrutamento.

Caraterísticas:

- Procurar emprego de forma simples e rápida.
- Pesquisas recentes são guardadas num histórico.
- Ver os novos empregos adicionados desde a última visita.
- Guardar ou enviar por e-mail as ofertas favoritas.
- Criar ou alterar o currículo.

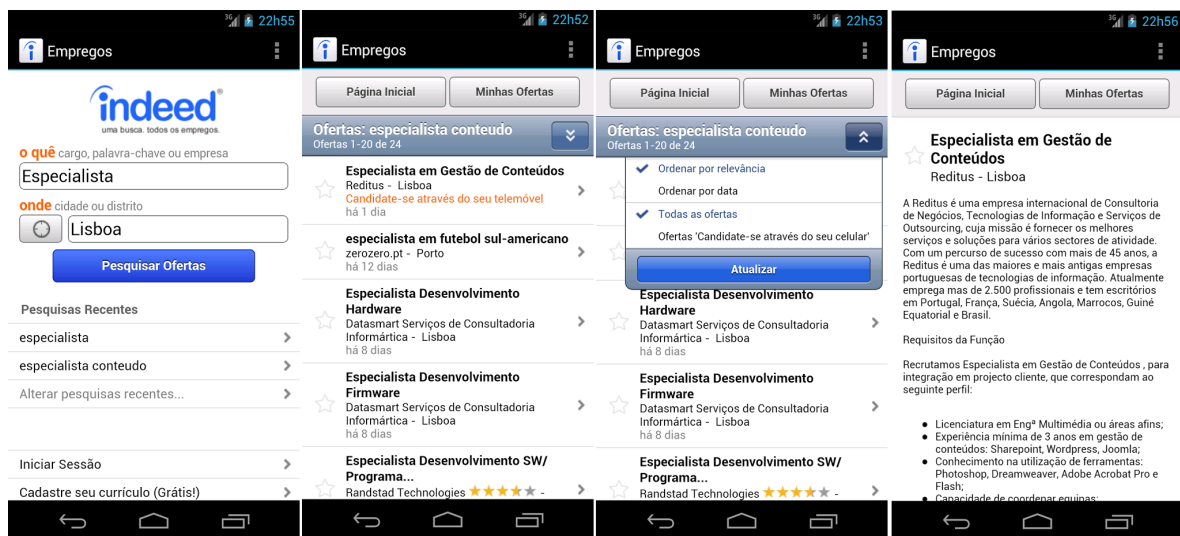


Figura 2.6: Imagens da aplicação Indeed.

2.4.2 Trovit Emprego

A aplicação Trovit Empregos permite pesquisar entre milhares de ofertas de emprego de diferentes sites na Internet. A aplicação permite fazer pesquisas por localização ou skills. Pode-se ainda usar os filtros de salário ou horário de trabalho para ajustar os resultados ao que realmente se está à procura. Podem-se criar alertas no correio eletrónico e partilhar os anúncios mais interessantes nas redes sociais diretamente da aplicação.

Caraterísticas:

- Ordena os resultados por salário, cidade, experiência, etc.
- Guarda as ofertas nos favoritos para compará-las melhor.
- Rápido acesso às ofertas de emprego mais recentes.

- Classifica as ofertas de emprego por relevância ou data.
- Partilha das ofertas de emprego mais interessantes via Facebook, twitter e Google+.
- Ofertas de emprego disponíveis em 38 países e 13 idiomas.

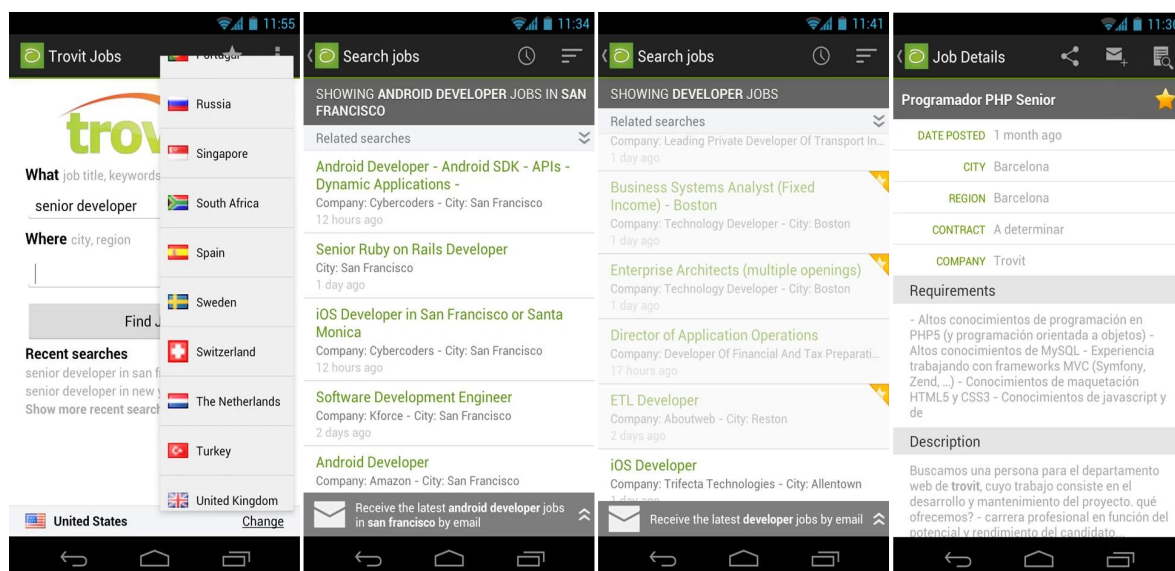


Figura 2.7: Imagens da aplicação Trovit.

2.4.3 Net-empregos

Aplicação não oficial para o portal de emprego net-empregos.com, vocacionada para encontrar ofertas de emprego em Portugal.

Caraterísticas:

- Pesquisa por zona geográfica e categoria profissional.
- Consultar o histórico de pesquisas recentes.
- Consultar anúncios de emprego com email de contacto incluído (para utilizadores registados).
- Partilha de anúncios para redes sociais, e-mail, SMS e outras aplicações.
- Possibilidade de guardar anúncios para consulta posterior.
- Pesquisas no contexto do anúncio (por empresa, zona e categoria profissional).
- Eliminar, individualmente ou todos, os anúncios guardados e/ou histórico de pesquisas.

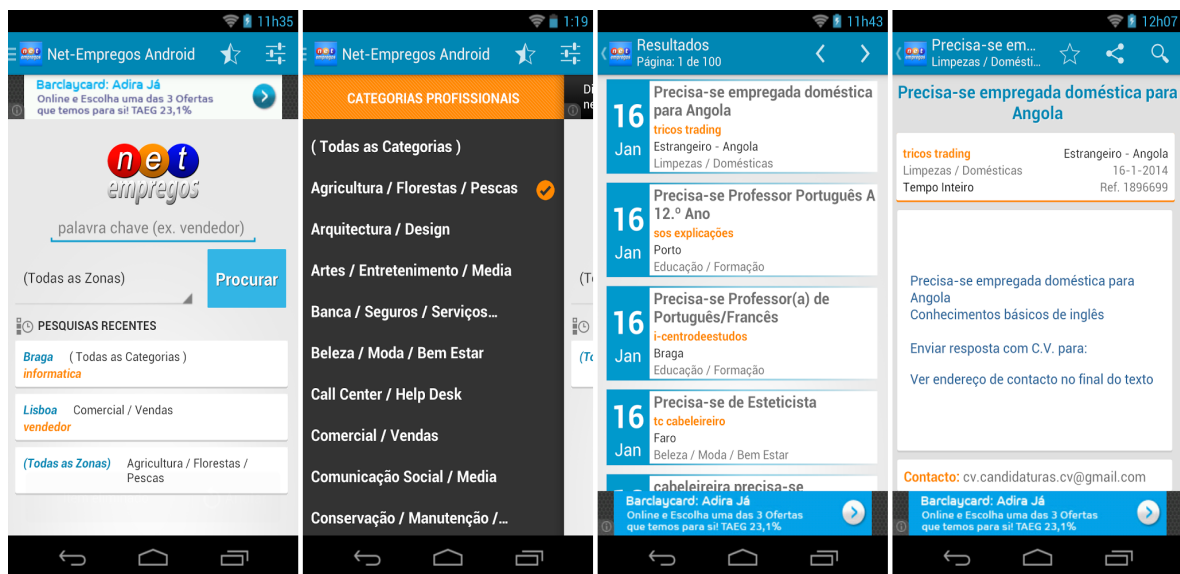


Figura 2.8: Imagens da aplicação Net-emprego.

Conclusão

As aplicações referenciadas são as melhores na sua área e para isso basta observar o número de utilizadores que cada uma tem. Apesar disso, estas ainda não chegam a todas as lojas de aplicações dos três sistemas operativos móveis mais destacados do mercado e tem algumas lacunas como é mostrado na Tabela 2.1.

Com a diversificação de redes sociais que existe nos dias de hoje, tanto de carácter lúdico como profissional onde as pessoas contemplam informação como áreas de estudo e as suas habilitações (skills), estas aplicações carecem dessa informação para dar as melhores e mais recomendadas ofertas de emprego para o perfil do candidato.

Quando um candidato encontra um emprego, este gosta de obter mais informação sobre a cidade ou país que futuramente o pode acolher e para isso existem plataformas que dão uma perspetiva do custo de vida da cidade ou país permitindo dessa forma perceber qual o devido valor da oferta.

Após esta análise surgiu a ideia de criar uma aplicação móvel que abrangesse todo o leque de funcionalidades que as demais oferecem e diferenciando oferecendo um conjunto características que as outras aplicações não oferecem.

Caraterísticas e Funcionalidades	Indeed	Trovit Empregos	Net-empregos
iOS	Sim	Sim	Não
Android	Sim	Sim	Sim
Windows Phone	Não	Não	Não
Número de utilizadores	>10 000 000	>100 000	>10 000
Procurar ofertas de emprego	Sim	Sim	Sim
Partilhar ofertas de emprego	Sim	Sim	Sim
Histórico de pesquisas	Sim	Sim	Sim
Procura automática de ofertas de emprego	Não	Não	Não
Ver últimas ofertas de emprego	Sim	Sim	Sim
Adicionar aos Favoritos	Sim	Sim	Sim
Conjunto alargado de países	Sim	Sim	Não
Ordenar ofertas de emprego	Sim	Sim	Não
Ofertas de emprego de vários países ao mesmo tempo	Não	Não	Não
Custo de vida da cidade da oferta de emprego	Não	Não	Não
Binário salário anual / custo de vida	Não	Não	Não
Análise de perfil do candidato	Não	Não	Não

Tabela 2.1: Tabela comparativa das aplicações de ofertas de emprego.

Capítulo 3

Metodologia e Requisitos

Esta secção tem como objetivo descrever o plano de estágio e a metodologia utilizada no desenvolvimento da aplicação móvel.

3.1 Metodologia

Com o desenvolvimento da aplicação móvel podem surgir algumas alterações aos requisitos levantados, alterações essas que se podem prender não só com o facto de qualquer projeto de software estar sujeito a essas alterações, mas também com a integração da aplicação móvel nos vários sistemas móveis o que torna evidente, a necessidade de utilizar uma metodologia ágil.

Todo o trabalho realizado irá utilizar o processo de desenvolvimento de software baseado na metodologia ágil Scrum, que é uma metodologia bastante popular no desenvolvimento iterativo e incremental. Uma vez que todo o projeto está integrado na equipa de desenvolvimento da plataforma JOObian, e esta ser a metodologia adotada pela a equipa.

Assim, poderá ainda referir-se o uso do JIRA¹, essencialmente através da utilização do registo de novas tarefas e funções de gestão do projeto.

Relativamente ao JIRA, a utilização de um quadro de tarefas traz vantagens uma vez que a velocidade de um projeto é fortemente determinada pelo entendimento geral da equipa do que está a acontecer. Assim, o quadro de tarefas permite a todos os elementos da equipa conhecer o progresso e quem está a trabalhar numa determinada tarefa e dessa forma, ter a noção global do projeto e do fluxo das várias tarefas. Um projeto baseado em Scrum é dividido em *sprints* (corridas) e em cada um destes é alocado o trabalho a realizar durante o período de tempo definido para os mesmos.

¹<https://www.atlassian.com/software/jira>

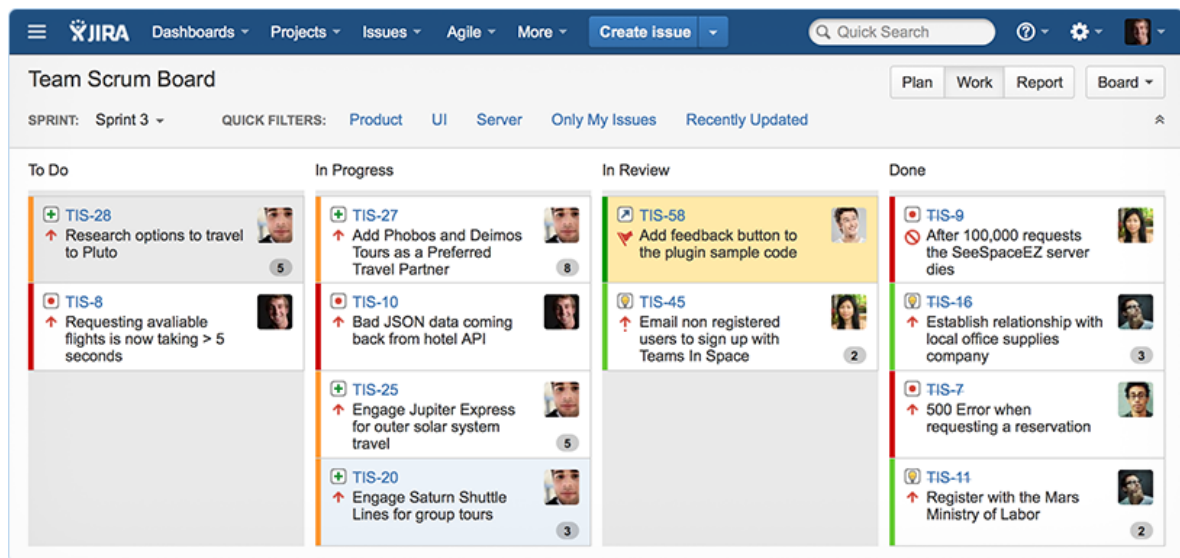


Figura 3.1: Imagem ilustrativa do quadro de tarefas da equipa.

3.1.1 Papéis

Existe uma equipa de desenvolvimento no qual o estagiário foi integrado, mas é bem patente a divisão de tarefas específicas do estagiário do resto da equipa responsável pela manutenção da plataforma JOObian.

Essa equipa está dividida em três papéis; em primeiro temos o cliente que tem que ser uma pessoa que tenha conhecimento do sistema e que represente a empresa para ajudar na tomada de decisões. A pessoa ideal para este papel é então o orientador do estágio por parte da empresa, visto ser ele quem tem mais conhecimentos do projeto e ao mesmo tempo representa os desejos e objetivos da empresa.

O *Scrum Master*, no seu papel original, é o responsável por manter o esforço da equipa no nível desejado, removendo impedimentos que surjam ao longo do progresso e mantendo o processo saudável. Neste cargo temos então a pessoa responsável pela equipa de desenvolvimento da plataforma JOObian.

Por último temos a equipa de desenvolvimento composta pelo responsável do back-end do JOObian, o responsável pelos testes de qualidade do software e o estagiário que fica responsável pelo desenvolvimento da aplicação móvel do JOObian.

Assim a equipa responsável é a seguinte:

- **Cliente:** Orientador por parte da empresa.
- **Scrum Masters:** Responsável pela equipa de desenvolvimento da plataforma JOObian.
- **Equipa desenvolvimento:** Composta por três elementos, responsável de back-end, responsável pelos testes de qualidade do software e o responsável pelo desenvolvimento da aplicação móvel do JOObian.

O papel de cliente e *Scrum Master* é minimizado, visto que toda a gestão dos artefactos e progressão do trabalho é responsabilidade do estagiário.

3.1.2 Artefactos

No começo de cada *sprint* é realizado um *Sprint Planning Meeting* no qual o cliente prioriza o *Product Backlog* e a equipa de desenvolvimento seleciona as tarefas que podem completar durante a *sprint* atual. Essas tarefas são então movidas do *Product Backlog* para o *Sprint Backlog*. [Eag08]

Temos ainda os gráficos de *Burndown*. O objetivo de ter este artefacto é manter um registo do progresso da implementação que permita a deteção de problemas de carácter consistente no seu decorrer e ajudar na sua correção.

Resumindo, os artefactos que serão utilizados são:

- *Backlog* do produto;
- *Backlog* do *sprint*;
- Gráfico de *Burndown*;

3.1.3 Atividades

As atividades que serão realizadas neste metodologia serão:

- Planeamento do *sprint* (corrida);
- Scrum diário;
- Revisão do *sprint*;
- Reunião com o orientador de estágio.

Planeamento da corrida consiste numa reunião com o cliente com o objetivo de definir quais as funcionalidades a ser implementadas durante a próxima corrida, tendo como resultado o *Backlog* da corrida e uma atualização do *Backlog do produto*.

O Scrum diário consiste numa pequena reunião, não mais de 15 minutos diários com o objetivo de sincronização da equipa. Será então o objetivo destas reuniões, entre o estagiário e *Scrum Master*, de fazer uma retrospectiva do que foi feito no dia anterior, o que vai ser feito nesse dia, e quais as dificuldades encontradas e como podem ser resolvidas.

No final de cada corrida, há uma reunião entre a equipa de desenvolvimento e o *Scrum Master* para que seja feita uma revisão do trabalho feito e fechar o gráfico de *Burndown*.

3.2 Definição de Requisitos

Nesta secção vão ser apresentados os atores que irão interagir com a aplicação e fazer utilização dos caso de uso do sistema. Também são apresentados os requisitos funcionais da aplicação móvel JOObian.

3.2.1 Atores e Pessoas

Para os atores são descritas algumas pessoas que ajudam a descrever a utilização que vai ser feita do sistema.

Para cada pessoa é apresentado:

- Descrição – Descrição sumária da pessoa(idade, nível de estudos, localidade, profissão/ área profissional).
- Objetivos pessoais – Os objetivos pessoais de cada pessoa que a podem levar a utilizar a aplicação.
- Objetivos para o sistema – Objetivos, tendo em conta os seus objetivos pessoais, que a pessoa terá ao utilizar a aplicação. Estes são descritos num pequeno texto e de seguida expostos os requisitos do sistema que este terá necessidade de utilizar.

3.2.2 Utilizador

Estes atores podem ser representados por três tipos de pessoas:

- **Candidato:** Utilizador, desempregado ou não, que se encontre no momento à procura de emprego.
- **Profissional:** Utilizador com uma carreira estável que queira estar mais informado sobre o mercado da sua área e obter informação que lhe permita continuar a construir a sua carreira.
- **Mundial:** Utilizador que pretende obter informações de oportunidades de emprego em vários países, que não se prende e que deseja obter as melhores oportunidades sem olhar a distâncias.

João Silva

Descrição

O João é um jovem de 23 anos, recém-licenciado em Marketing pela Universidade de Aveiro.

Objetivos pessoais

No fim do curso decidiu saber a oferta atual de trabalho na sua área para assim tomar a decisão de continuar os seus estudos e saber melhor que tipo de formação é mais pedida, ou, no caso de encontrar ofertas de trabalho interessantes, tentar a sua sorte numa destas e fazer a sua entrada no mercado de trabalho.

Objetivos para o sistema

A aplicação vai dar acesso ao utilizador a uma lista de ofertas recomendadas tendo em conta a informação do seu perfil. Pode também pesquisar por ofertas de emprego na sua área e consultá-las, bem como saber a formação recomendada consoante o emprego em que esteja interessado.

Requisitos:

- Ver sugestões de emprego;
- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação da oferta de emprego;
- Ver formação recomendada.

Joana Soares

Descrição

A Joana tem 27 anos, é natural de Faro e tirou Mestrado em Engenharia Informática na Faculdade de Ciências e Tecnologia da Universidade de Coimbra e encontra-se atualmente a trabalhar numa empresa de desenvolvimento de software na mesma cidade.

Objetivos pessoais

O seu posto de trabalho permite-lhe ter uma vida estável, mas devido à distância que a separa da família e dos amigos, num futuro próximo a Joana gostava de encontrar trabalho mais próximo da sua terra natal. No entanto gostaria de, pelo menos, manter um salário que lhe permita continuar a ter o estilo de vida atual no seu novo posto de trabalho.

Objetivos para o sistema

Ao utilizar a aplicação a Joana pode procurar ofertas de emprego na sua área profissional e localização pretendida. Pode ainda consultar uma lista de empregos recomendados consoante a sua formação e localização. Para ter uma ideia do salário que poderá ficar a ganhar se, se mudar para estar mais perto da casa pode também consultar métricas que lhe mostrem uma

média quanto ganha de quanto ganha alguém nessas condições.

Requisitos:

- Ver sugestões de emprego;
- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação da oferta de emprego;
- Ver as habilitações em falta no seu perfil;
- Ver métricas de ajuda à gestão de carreira.

Tiago Santos

Descrição

O Tiago tem 32 anos e trabalha à 6 anos como web developer na mesma empresa.

Objetivos pessoais

Ele gosta do seu posto de trabalho na empresa mas ao mesmo tempo sente que não está a progredir na sua carreira. Decide então procurar habilitações (skills) que o ajudem a alcançar a chegar a gestor de projetos na sua área.

Objetivos para o sistema

Ao utilizar a aplicação o Tiago pode procurar por ofertas de emprego semelhantes ao cargo que ele quer atingir e ter acesso as sugestões de que tipo de formação e habilitações são necessários para atingir um cargo deste tipo.

Requisitos:

- Ver sugestões de emprego;
- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação da oferta de emprego;
- Ver formação recomendada.

João Martins

Descrição

O João tem 33 anos e trabalha à 8 anos como Engenheiro Eletrotécnico já passou por várias empresas em diversas cidades e países da Europa.

Objetivos pessoais

Ele gosta dos desafios e tem uma necessidade de crescer profissionalmente e salarialmente, gosta de conhecer novas cidades e culturas. No entanto decidiu que chegou uma nova etapa e decide procurar melhorar não só a sua condição salarial mas também a qualidade de vida/-poder de compra e necessita de saber qual será não só o próximo emprego mas também a sua próxima cidade/país

Objetivos para o sistema

Ao utilizar a aplicação este utilizador pode obter recomendações de ofertas de emprego direcionadas e alinhadas com o seu perfil, sendo que as mesmas deverão ter em conta as restrições por ele impostas, nomeadamente a maximização do binário salário/custo de vida.

Requisitos:

- Ver sugestões de emprego;
- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação da oferta de emprego;
- Ordenar as ofertas de emprego por (Salário e/ou Índice Custo de Vida).
- Ver índice de qualidade de vida.

3.3 Requisitos Funcionais

Para definir os requisitos foram utilizados casos de uso. A preferência destes as *user stories* teve a ver com o nível de detalhe que um caso de uso permite ter de um requisito. Tendo este nível de detalhe, qualquer pessoa que vá implementar o sistema, ao ler os requisitos, conseguirá ter uma ideia melhor do que é pretendido implementar. Utilizando este tipo de requisitos podem também ser elaborados diagramas de casos de uso que permitem uma melhor visão do sistema no seu todo e como cada requisito interage entre si. Para cada caso de uso são definidos os seguintes pontos:

- **ID do requisito:** Número de identificação do requisito.
- **Prioridade:** Classificação da prioridade do requisito, segundo os critérios: Must (prioridade máxima), Should (prioridade média), Nice (prioridade baixa).
- **Dependências:** ID's dos requisitos dos quais o requisito em questão está dependente.
- **Descrição:** Descrição da funcionalidade do requisito.

No Anexo A é apresentada a lista de casos de uso. O documento de requisitos desenvolvido, é um documento que a empresa usa internamente e encontra-se nos anexos do relatório. Neste são apresentados todos os casos de uso descritos em pormenor.

De seguida vão ser mostrados todos os requisitos que fazem parte da aplicação e a sua prioridade no desenvolvimento.

3.3.1 Requisitos contas de utilizador

ID	Caso de Uso	Prioridade
0	Autenticar na aplicação através do LinkedIn	MUST
1	Autenticar na aplicação através do processo interno	MUST
2	Redirecionar para adicionar uma conta no LinkedIn	MUST

Tabela 3.1: Lista de casos de uso de utilizador.

3.3.2 Requisitos gerais da aplicação

ID	Caso de Uso	Prioridade
0	Ver sugestões de ofertas de emprego	MUST
1	Ordenar ofertas de emprego por relevância, proximidade e data	MUST
2	Ver perfil do utilizador	MUST
3	Procurar ofertas de emprego	MUST
4	Filtrar pesquisa de ofertas de emprego	MUST
5	Ver informação das ofertas de emprego	MUST
6	Ver as suas ofertas de emprego favoritas	MUST
7	Escolher os países do qual quer receber ofertas de emprego	MUST
8	Receber notificações de novas ofertas de emprego	SHOULD

Tabela 3.2: Lista de casos de uso da aplicação.

3.3.3 Requisitos da oferta de emprego

ID	Caso de Uso	Prioridade
0	Ver as skills que fazem match	MUST
1	Ver as skills em falta	MUST
2	Ver as skills de idiomas	MUST
3	Ver oferta salarial	NICE
4	Adicionar ofertas de emprego aos favoritos	MUST
5	Remover oferta de emprego dos favoritos	MUST
6	Ver informação das ofertas de emprego	MUST
7	Ver oferta de emprego no Indeed	MUST
8	Adicionar skills em falta ao perfil	MUST
9	Partilhar oferta de emprego (e-mail, Facebook, twitter, Google+, etc...)	NICE

Tabela 3.3: Lista de casos de uso da oferta de emprego.

3.3.4 Requisitos da pesquisa

ID	Caso de Uso	Prioridade
0	Pesquisar por skills, países ou empresas	MUST
1	Ver ofertas de emprego por relevância, proximidade e data	MUST
2	Ver resultados da pesquisa por oferta de emprego recomendadas	MUST
3	Ver todas as ofertas de emprego	MUST

Tabela 3.4: Lista de casos de uso da pesquisa.

3.3.5 Requisitos da estatística

ID	Caso de Uso	Prioridade
0	Ver o número de ofertas avaliadas do conjunto de países em seleção.	MUST
1	Ver as skills que mais match fizeram	MUST
2	Ver as skills que mais estão em falta no perfil do candidato	MUST
3	Ver as localizações que mais ofertas têm	MUST
4	Ver as empresas que mais ofertas de emprego colocam	MUST

Tabela 3.5: Lista de casos de uso da pesquisa.

3.3.6 Requisitos do perfil

ID	Caso de Uso	Prioridade
0	Ver experiência profissional	MUST
1	Ver idiomas	MUST
2	Ver skills	MUST
3	Ver carreira acadêmica	MUST
4	Adicionar novas skills	MUST
5	Importar CV	NICE

Tabela 3.6: Lista de casos de uso do perfil.

3.3.7 Requisitos de interface externa

ID	Caso de Uso	Prioridade
0	Conetividade GPS	MUST
1	Conetividade Wi-fi	MUST
2	Conetividade 3G	MUST
3	Recolher dados através da API do LinkedIn	MUST

Tabela 3.7: Lista de casos de uso da interface externa.

3.3.8 Requisitos de performance

ID	Caso de Uso	Prioridade
0	Cache das ofertas de emprego para visualização offline	MUST
1	Cache de conteúdo estático	MUST

Tabela 3.8: Lista de casos de uso de performance.

3.3.9 Requisitos do sistema

ID	Caso de Uso	Prioridade
0	Recuperação do sistema em caso de falha	MUST

Tabela 3.9: Lista de casos de uso do sistema.

Capítulo 4

Arquitetura e desenho

Nesta secção será apresentada a arquitetura do sistema implementado que dá suporte à aplicação móvel e a arquitetura no qual a aplicação está assente. O JOObian como aplicação móvel tem que deixar toda a parte de recolha de informação, processamento e tratamento dos dados do lado do servidor. Numa primeira etapa é dada uma visão lógica do sistema em que podemos ver os vários componentes que constituem o sistema no seu geral, assim como a comunicação entre as suas interfaces, em vários níveis. Do ponto de vista físico a plataforma é constituída apenas por um servidor, encontrando-se a grande complexidade da arquitetura no modelo lógico. Devido à necessidade que esta plataforma tem da existência de vários módulos, para que num futuro se possam adicionar novos componentes, foi seguido o padrão de uma arquitetura orientada a serviços.

4.1 Arquitetura Servidor

4.1.1 Plataforma JOObian

A plataforma JOObian irá fornecer serviços de apoio ao cliente móvel. Para entender melhor a plataforma JOObian iremos dar uma visão geral desta plataforma de recrutamento que terá então que ser capaz de interagir e obter dados de várias fontes externas.

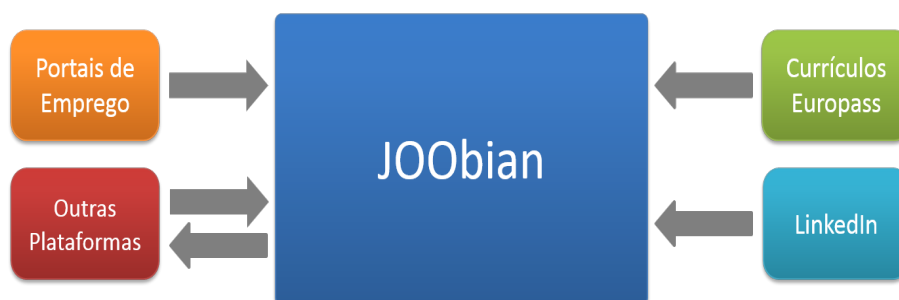


Figura 4.1: Vista global da plataforma e conexões com fontes externas.

Como podemos identificar na figura acima, há quatro fontes que poderão ser utilizadas para introdução direta ou indireta de dados no sistema.

- Portais de emprego: Estes serão utilizados para obter as últimas ofertas publicadas e possíveis de obtenção por feeds RSS.
- Currículos Europass: Para facilitar o registo dos dados curriculares do utilizador, este poderá utilizar um currículo, formato Europass, para preencher campos do formulário de registo.
- LinkedIn: Em vez do currículo, descrito acima, um utilizador pode também utilizar a informação do seu perfil do LinkedIn para o mesmo efeito, pois este funciona como um currículo *online*.
- Outras Plataformas: Pode também haver integração com outras plataformas de modo a permitir que empresas que já utilizem outros sistemas possam utilizar serviços disponibilizados pela nossa plataforma.

A vantagem de ter duas fontes de obtenção de informação para preenchimento de dados do utilizador, é a existência de possíveis utilizadores terem apenas um dos métodos, assim a probabilidade de um destes conseguir importar os dados de outro sítio onde já estejam preenchidos é maior.

De seguida é explicado em detalhe as quatro fontes utilizadas no nosso sistema:

Feeds RSS:

Os portais de emprego identificados que poderão ser utilizados como fontes de feeds RSS foram o Sapo emprego, o Expresso emprego e o ITJobs. Como estes feeds RSS devolvem um ficheiro XML e para cada uma das fontes o esquema é diferente, é necessário considerar a utilização de uma configuração para cada fonte assim como a possibilidade de alteração do esquema por questões de manutenção e adição de novas fontes. Para isto, deve ser possível configurar genericamente qualquer fonte de feeds RSS que se pretenda utilizar.

Como vão sendo lançadas novas ofertas de emprego nos feeds RSS, esta extração deve ser feita periodicamente de maneira a que estas ofertas sejam adicionadas à plataforma.

Currículos Europass:

Estes vão ser utilizados na altura do registo ou edição de um perfil de utilizador para ajuda no preenchimento do formulário do perfil. É necessário que o utilizador carregue o ficheiro para a plataforma, que terá que ler o mesmo e preencher os campos do formulário com a informação que for possível extrair.

É necessário considerar as várias versões existentes do formato Europass, pois cada versão tem um esquema e modelo de dados diferente e tem que ser tratada de maneira diferente, assim como várias línguas. Também aqui terá que haver uma configuração que permita definir isto.

LinkedIn:

Os dados de perfil do LinkedIn, tal como os currículos, podem ser utilizados no registo e edição de um perfil de candidato. Este acesso é feito através da API do LinkedIn. Para isto é necessário ter uma aplicação registada no site de programador do LinkedIn¹, pois é preciso as credenciais obtidas neste registo para aceder à API. É ainda necessário que o utilizador

¹<https://developer.linkedin.com>

que queira utilizar este método de extração conceda autorização para tal, o que resulta na obtenção de um *access token* que é utilizado para aceder aos dados do perfil do mesmo.

Como após ter o *access token* do utilizador é possível aceder à informação do seu perfil posteriormente, a não ser que expire a sua validade ou o utilizador revogue a autorização dada, é pretendido fazer uma atualização periódica destes dados de modo a que o utilizador, ao atualizar o seu perfil no LinkedIn, este seja automaticamente atualizado na plataforma.

Outras Plataformas:

Para questões de integração do sistema com outras plataformas que os clientes utilizem, será disponibilizado uma API sob a forma de um serviço RESTful. Este serviço devolverá respostas em JSON e permitirá que os clientes da plataforma tenham ao seu dispor as mesmas funcionalidades que teriam acedendo à aplicação web OObian, a não ser que tal não seja possível por questões de visualização.

Além destas funcionalidades descritas o sistema deve disponibilizar consulta de ofertas de emprego e perfis de candidatos através de pesquisa, por palavras-chave e filtros, e recomendações geradas através de análise de dados.

4.1.2 Plataforma Funcional e Lógica do JOObian

De uma perspetiva funcional o JOObian é composto por três grandes grupos:

- Clientes
- Servidor
- Sistemas Externos

Os Clientes são responsáveis pela apresentação e interação entre os dados existentes no servidor e o utilizador, os clientes poderão ser móveis: Aplicações para iOS, Android, Windows Phone e outros dispositivos móveis; ou então clientes HTML (página Web típica). Neste grupo enquadra-se o desenvolvimento das aplicações móveis detalhadas nos documentos de requisitos e especificação existentes no Anexo A.

O Servidor é a peça fulcral neste sistema, fornecendo serviços para os clientes, integrando dados de sistema externos, atualizando, gerindo, catalogando e associando ofertas de emprego com candidatos. Para isto foi desenvolvido uma aplicação Java EE sob o formato de um EAR que permitiu desenvolver todas as funcionalidades de suporte e empacotar as mesmas sendo disponibilizadas num servidor aplicacional Java EE que no nosso caso é o JBoss² 6.1 EAP.

Os Sistemas externos são muito importantes para efetuar um conjunto de sub-tarefas, nomeadamente a integração dos dados e de autenticação.

Passamos para uma visão integrada com todas as *layers* acima referidas, nomeadamente o Cliente, Servidor e Sistemas Externos.

²<http://www.redhat.com/products/jbossenterprisemiddleware/application-platform/>

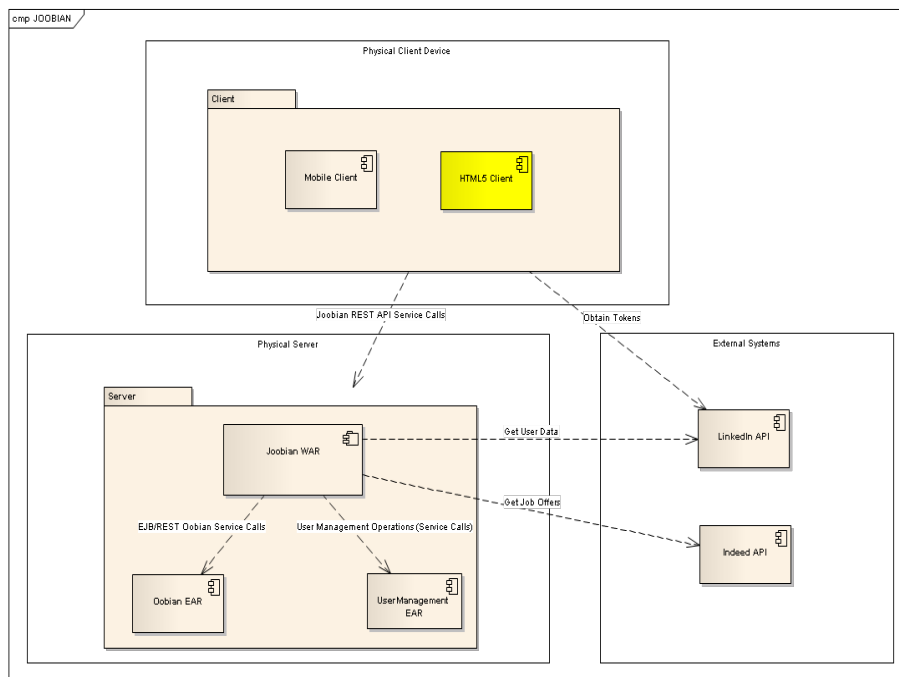


Figura 4.2: Visão birds-eye do JOObian.

Passa-se a explicar cada uma das componentes que integram o Servidor:

Jobian WAR:

Componente responsável por fornecer todos os serviços necessários aos clientes JOObian. Este componente está empacotada sob a forma de um Java Web Enterprise Application (WAR package).

User Management EAR

Componente responsável por operações de registo, login, validação de tokens e reset password. Permite obter tokens de forma a assegurar a segurança/acesso entre os diferentes componentes e serviços.

Oobian EAR

Componente responsável por tarefas de indexação, mapeamento, recomendações, anotação semântica entre outros. Fornece também serviços de apoio ao JOObian, tipicamente referentes a operações de indexação, mapeamento de dados, anotações semânticas e recomendações de ofertas de emprego.

Plataforma Oobian:

O Oobian é uma plataforma horizontal de gestão de conhecimento empresarial. Após 4 anos de desenvolvimento, pela Maisis - Information Systems, Lda, foi lançada como produto em 2009 e é atualmente o produto principal da empresa.

O principal foco desta plataforma é dar às organizações uma ferramenta que consiga pesquisar, organizar e gerir de uma forma lógica e fácil todo o tipo de documentos em formato digital para estas conseguirem tirar o máximo partido da informação que possuem. Para isto o Oobian tem uma interface web, que permite ver a informação em forma de grafo de

conhecimento, fazer *drill up* e *drill down*, pesquisa e desambiguação de conceitos, filtragem de resultados, georreferenciação de entidades no mapa, etc. . .

A plataforma foi desenvolvida para facilmente integrar com qualquer sistema que as organizações possam possuir, utilizando SOA e REST, pode ser integrada com todo o tipo de base de dados e vários repositórios de conteúdo. A interpretação semântica de conteúdo é baseada em ontologias, permitindo uma forte adoção de modelos de negócio e suporta as recomendações de web semântica do W3C, como OWL e RDFS.

Como este permite a fácil integração com serviços externos e a importação de modelos de conhecimento em OWL, foi escolhido para a navegação nos dados a utilização de um modelo de conhecimento que permite fazer mais facilmente a análise dos dados extraídos.

De seguida vamos passar a explicar alguns dos detalhes e dependências do JOObian, nomeadamente o consumo de serviços Oobian, esta dependência é justificada pela capacidade que o Oobian tem de efetuar integração e mapeamento de dados, efetuar pesquisas, indexar, gerir e mostrar informação, sendo que o ponto crítico é a capacidade de obtenção de recomendações orientadas ao utilizador e a capacidade de efetuar amotações semânticas de extrema velocidade. Este último ponto é vital para poder associar habilitações (skills) às ofertas de emprego, isto é feito através de serviços do Oobian que associam a entidade Skill a um texto, neste caso o texto da descrição das ofertas de emprego.

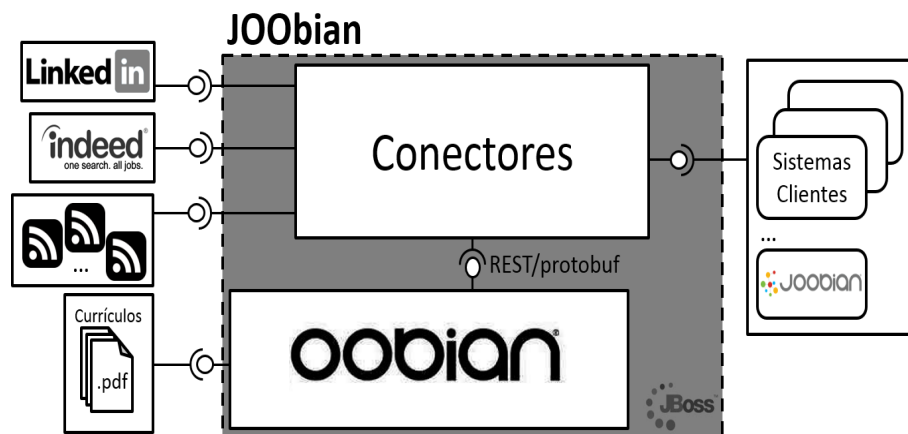


Figura 4.3: Conetores do JOObian.

Note-se que na Figura 5.3 podemos ver o Oobian como dependência do JOObian, esta dependência pode ser realizada de pelo menos duas formas diferentes, dependendo da estratégia de *deployment* montada, podemos usar o consumo de serviços SOAP e REST (JSON e Protobuf³), ou em alternativa usar os mecanismos nativos dos EJB com a declaração e consumo das interfaces @Local ou @Remote.

No próximo diagrama é uma evolução do anterior no qual mostramos com mais detalhe os principais componentes existentes em cada um destes sistemas.

³<https://code.google.com/p/protobuf/>

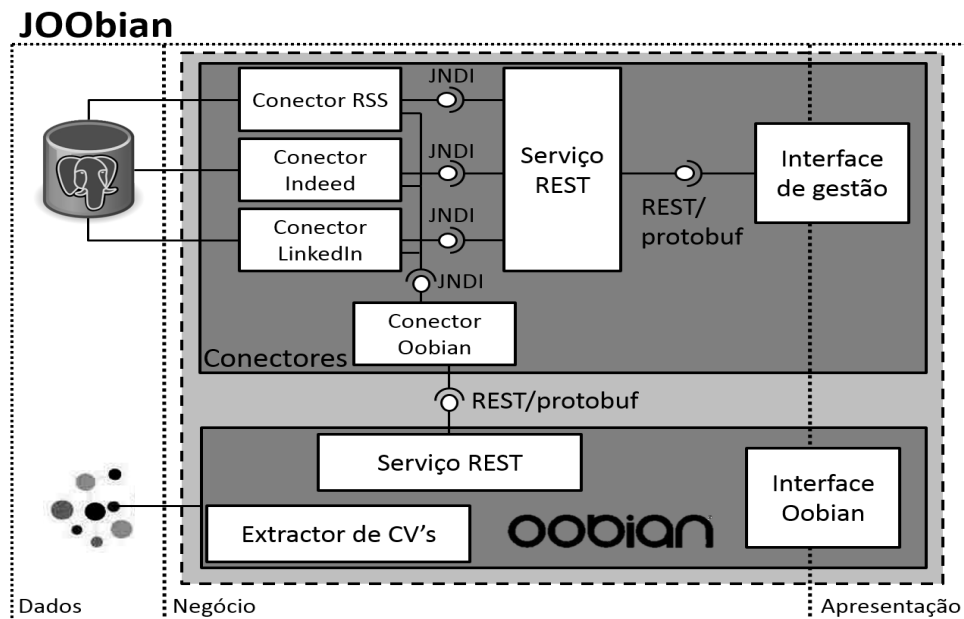


Figura 4.4: Componentes do JOObian.

Conetor RSS, Indeed e LinkedIn: Estas são os conetores responsáveis por extrair a informação das várias fontes de dados já descritas. Cada um dos conetores disponibiliza uma interface que pode ser utilizada pelo serviço REST para chamar os métodos que disponibiliza. Para além disto utiliza o módulo “Conetor Oobian” para fazer a importação de dados no modelo de conhecimento Oobian.

Conetor Oobian: O conetor que consome serviços REST do Oobian é o responsável por mapear os objetos utilizados internamente nos conetores para armazenar e tratar a informação em instâncias de conhecimento e utilizar esses serviços do Oobian para enviar estas mesmas instâncias. Este utiliza também os serviços do Oobian que permitem fazer análise de informação e criar associações de conhecimento a instâncias de ofertas de emprego e pessoas.

Serviço REST: O serviço REST permite disponibilizar serviços de gestão, que são utilizados pela interface de gestão dos conetores. Este também é responsável por fornecer às aplicações clientes o acesso a serviços. Enquanto que nos serviços internos é utilizado o Protobuf para comunicação, nos serviços externos é utilizado JSON.

Interface Gestão: A interface de gestão é uma interface que permite realizar tarefas de gestão dos conetores como inicialização destes e adição e agendamento de queries a ser feitas às APIs.

4.1.3 Perspetiva Física

De seguida vamos ilustrar esta plataforma sobre uma perspetiva física, tendo também a atenção especial para as questões de escalabilidade e de que forma isso poderá afetar o desenvolvimento da aplicação tendo impacto não só na sua estrutura física mas também na

arquitetura lógica.

Resumidamente o próximo diagrama responde às questões de desenvolvimento: “Quais os dispositivos/entidades físicas necessárias neste sistema?”; “Quantos Servidores?”; “Onde vão estar?... Na cloud?”; “Como comunicam estas entidades?”; “Temos sistema para que de um dia para o outro sejam suportados 1000 novos utilizadores?”.

O ecossistema JOObian necessita de clientes (ex: clientes móveis) que irão ser dispositivos físicos que poderão estar dispersos pelo mundo inteiro. Necessita também de um ou mais servidores, que poderão estar alojados nas instalações da Maisis - Information Systems, Lda, ou num serviço modelo de Cloud Hosting, bastando para isso usar a plataforma OpenShift⁴ que permite ter JBoss num modelo cloud.

Mais importante que este ponto é a vantagem que existe de ter o JOObian separado logicamente e dependendo do desenvolvimento, podemos também ter uma separação física. Esta separação permite que em situações de elevada necessidade de escalabilidade não só o JOObian possa existir com redundância (em múltiplos servidores) como também o próprio Oobian poderá ter esta capacidade com configurações e capacidades diferentes, ou seja podemos ter três máquinas com o JOObian e vinte máquinas com o Oobian, uma vez que o processamento mais pesado é feito pelo Oobian, podendo usufruir ainda do benefício de múltiplos *mirrors* ou *shards* do índice do Lucene⁵. [Bru13]

⁴<http://www.jboss.org/openshift.html>

⁵<http://lucene.apache.org/core/>

JOOBIAN-SCALABLE

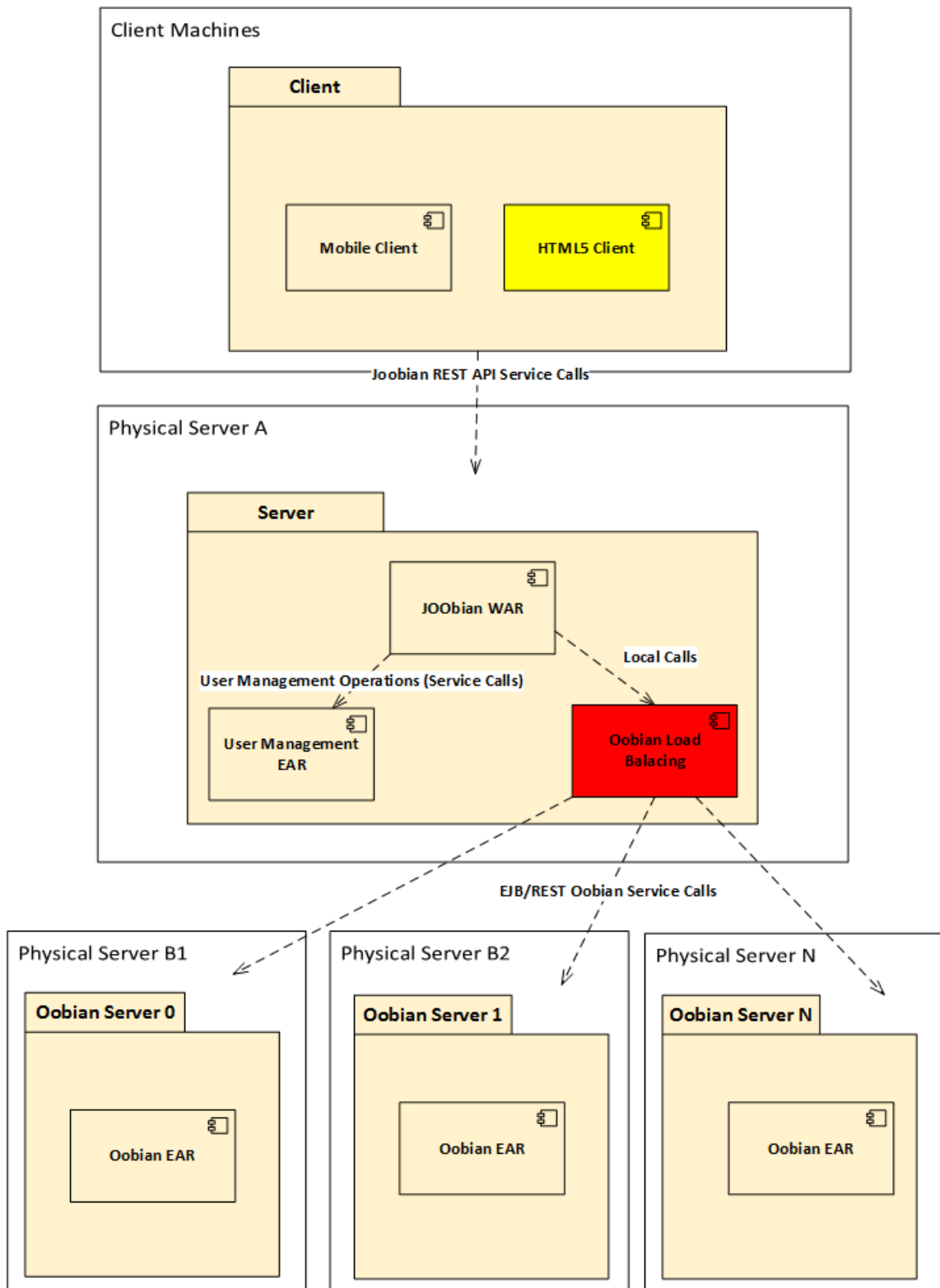


Figura 4.5: Diagrama físico com especial atenção à escalabilidade.

4.2 Arquitetura PhoneGap

Imagem seguinte mostra como é que uma aplicação PhoneGap trabalha, isto é, como é que a componente JavaScript é capaz de comunicar com a API nativa e vice-versa.

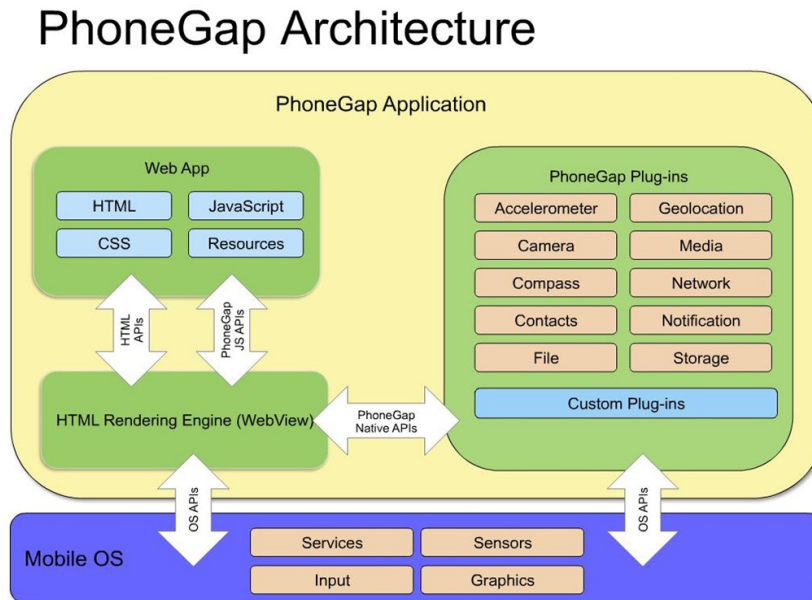


Figura 4.6: Diagrama no qual está assente um aplicação híbrida PhoneGap.

Todas as aplicações PhoneGap tem os seguintes componentes:

- Um navegador *chromeless*. Em iOS e Android chama-se WebKit (UIWebView no iOS e WebView no Android).
- *Bridge* entre JavaScript e Nativo que permite a comunicação entre a aplicação HTML e a plataforma nativa.
- *Bridge* entre Nativo e JavaScript que permite a plataforma nativa comunicar com a aplicação HTML.

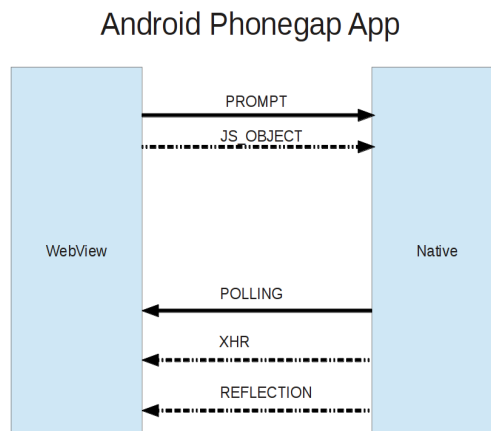


Figura 4.7: Diagrama fluxo de dados no Android entre a WebView e a parte Nativa.

No Android por definição a *bridge* entre o JavaScript e a parte Nativa é chamado de *Prompt*. As funções JavaScript (como câmara, contactos, etc...) são convertidos pelo PhoneGap em comandos para a *Prompt* e interceptados pela WebView e com base numa assinatura específica chama o respetivo plug-in nativo (câmara, contactos, etc...).

Outra forma de comunicação entre a parte nativa e o JavaScript é através de JS_Object. Quando o WebView é carregado e a ligação com JavaScript está definida, o WebView adiciona uma interface de JavaScript que chama um objeto Java (chamando o respetivo plug-in nativo com base nos argumentos da interface).

A ligação que permite a parte nativa comunicar com o JavaScript, por definição chama-se *Polling* e o JavaScript a cada 50 milissegundos faz uma chamada à parte nativa. Isto não é a solução ideal, mas é a solução que funciona no maior número de dispositivos e na maioria das configurações.

Outra ligação nativa com o JavaScript é o XHR (o chamado HANGING_GET, internamente é uma referência às ligações XHR de longa vida). É executado um servidor de resposta a nível local e responde às solicitações XHR.

Existe ainda uma outra ligação que usa Java Reflection sobre o WebView para chamar métodos sobre uma classe Java, mas só está disponível nas versões de Android superiores a Android 3.2.

Em seguida irá ser analisada como as aplicações PhoneGap trabalham no iOS.

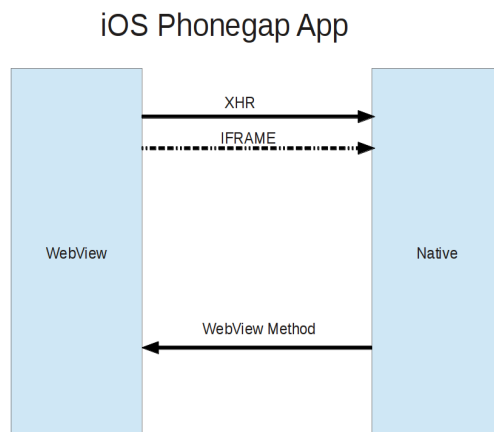


Figura 4.8: Diagrama fluxo de dados no iOS entre a WebView e a parte Nativa.

Comparando com o Android o iOS tem menos ligações. A *bridge* entre JavaScript e a parte Nativa são feitos através de uma iframe. As chamadas do JavaScript são armazenadas numa fila, que são lidas e executados pela parte nativa.

No iOS o XHR faz pedidos a uma URL falsa com os comandos no cabeçalho. Estes comandos são interceptados, serializados e depois executados.

De nativo para JavaScript, as aplicações PhoneGap em iOS têm apenas uma ligação e toda a comunicação é feita através de um método chamado `UIWebView stringByEvaluatingJavaScriptFromString`.

Com o conhecimento sobre como as aplicações PhoneGap trabalham, é possível escrever mais e melhores aplicações bem como plugins e otimizar o desempenho da mesma.[The12]

4.3 Estrutura da aplicação móvel

Como já referido anteriormente, a aplicação foi desenvolvida num sistema híbrido, e recaiu sobre a versão igual ou superior a 4.0 no sistema Android e sobre a versão igual ou superior a 6 no iOS. A escolha destas versões recaiu sobre o facto de representar 84,3%[And14] dos utilizadores no Android e 98%[App14] no iOS.

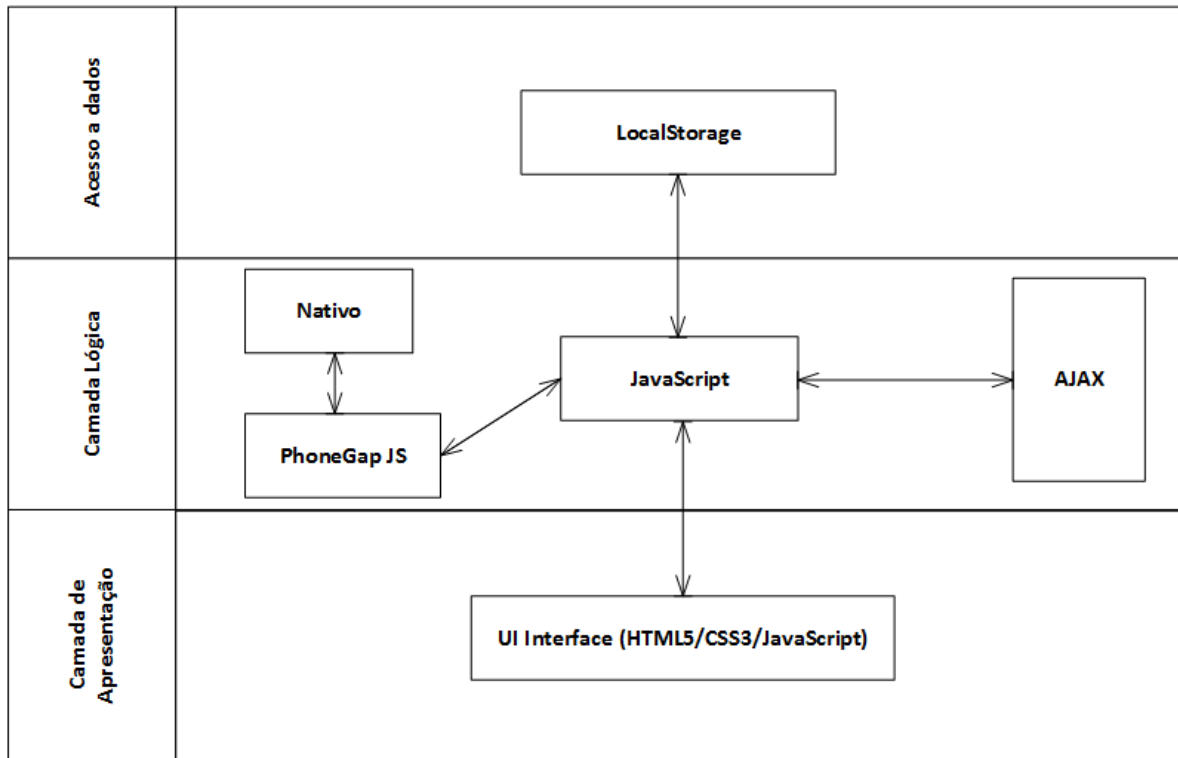


Figura 4.9: Solução da implementação da aplicação móvel JOObian.

A Figura 5.9 ilustra a solução no qual esta assente a aplicação móvel JOObian. Esta está dividida em três camadas distintas. Apesar de não existir um padrão de software esta visualização é simples de entender e permite uma explicação detalhada do desenho da aplicação. A primeira camada que refere-se à localização onde aplicação guarda os dados mais importantes. Podemos dividir esta camada em outra duas: persistente e não persistente. Aqui só é mostrada a parte persistente por uma questão de coerência pois os dados não persistentes apenas existem em execução da aplicação. A camada lógica é responsável pela apresentação correta dos dados na interface do utilizador. Por ultimo temos a camada de apresentação que é responsável por mostrar os dados da camada lógica.

4.3.1 Camada de dados

Como já se referiu, a camada de dados, controla todos os dados que estão armazenados ou são utilizados na aplicação. Podemos dividir estes dados em persistentes, estes permanecem

em *cache* mesmo após a aplicação ser terminada. Estes referem-se a dados de login e tokens, às ofertas de emprego do utilizador, favoritos, perfil e definições, no qual ficam armazenados em `LocalStorage` que é uma das novas funcionalidades do HTML5. Este armazenamento dos dados por parte da aplicação cliente permite ao utilizador trabalhar *offline*, quando não existe conectividade com o servidor. Os dados não persistentes estão apenas disponíveis durante o tempo de execução da aplicação.

4.3.2 Camada lógica

Esta representa toda a lógica da aplicação, sendo aqui que se encontra a maior parte do código desenvolvido.

Todas as páginas HTML tem um controlador, isto é, existe um ficheiro JavaScript que contem toda a lógica daquela página, como por exemplo: validação de dados introduzidos pelo utilizador, pedidos de informação ao servidor, tratamento dos dados recebidos do servidor, entre outros.

Relativamente a bibliotecas externas podem-se destacar:

- **analytics.js**⁶ é uma ferramenta da Google que permite uma avaliação completa de todo o percurso do utilizador em termos da aplicação, desde a descoberta, passando pela transferência e até à relação estabelecida. Os relatórios são personalizados para programadores de aplicações para dispositivos móveis e profissionais de marketing e transmitem-lhes as informações que lhes interessam.
- **md5.js**⁷ É uma biblioteca que disponibiliza uma função de *hash* MD5. Tem sido utilizada numa variedade de aplicações com necessidade de segurança e também é comumente usado para verificar a integridade dos arquivos. Embora, o MD5 não seja de colisão resistente, também não é adequado para aplicações com certificados SSL ou assinaturas digitais.
- **OAuthSimple.js**⁸ realiza a interação com os módulos OAuth indispensáveis para a comunicação com o LinkedIn.

Os serviços são componentes que realizam longas operações em segundo plano e não possuem uma interface própria para o utilizador.

O PhoneGap JS faz a ponte entre a parte nativa e é utilizado aqui para obter a localização do dispositivo, o identificador único do dispositivo para a plataforma JOObian enviar uma notificação com o número de ofertas novas desde a última vez que o utilizador acedeu à aplicação. Para informar o utilizador de alguma ação que este tenha feito ou de falha da aplicação é mostrado um alerta nativo, por último na partilha da oferta de emprego também é mostrada uma interface nativa com todas as aplicações que o utilizador pode partilhar a oferta.

4.3.3 Camada de representação

Esta camada define a estrutura visual da aplicação, sendo um dos pontos críticos da aplicação, apesar de ter bom conteúdo uma interface pouco cuidada e confusa é um dos principais pontos de fuga dos utilizadores.

⁶<https://developers.google.com/analytics/devguides/collection/analyticsjs/?hl=pt-PT>

⁷<https://code.google.com/p/crypto-js>

⁸<https://github.com/jrconlin/oauthsimple/tree/master/js>

As páginas são construídas em HTML e os estilos aplicados a cada um dos elementos pode ser feito de duas formas, declarando o estilo no próprio HTML ou criando classes no CSS e desta forma deixando mais limpo e simples de entender o HTML.

Capítulo 5

Desenvolvimento, Integração e Testes

Neste capítulo são apresentados os vários componentes que foram implementados para construir e montar toda a aplicação desenhada e exposta até agora. Este encontra-se dividido em secções que dividem todo o processo de desenvolvimento. Em cada uma das secções são descritos todos os passos tomados durante a fase de implementação, incluindo as principais dificuldades encontradas e estratégias para mitigar as mesmas.

A aplicação móvel JOObian nasceu com o objetivo principal de facilitar a obtenção de ofertas de emprego relevantes por parte de um determinado utilizador de uma forma automática. Para alcançar este objetivo considerou-se que a aplicação deveria ser de fácil uso, sendo que a mesma deveria ser disponibilizada em dispositivos móveis para os sistemas operativos mais relevantes (iOS, Android).

Após uma análise cuidada, definida no capítulo 2, optou-se por desenvolver a aplicação usando uma abordagem híbrida (PhoneGap).

Quanto às ferramentas de apoio ao desenvolvimento desta aplicação optou-se por usar as seguintes:

- **Ambientes de Desenvolvimento:** Como ferramentas de desenvolvimento foram usados dois IDEs o Eclipse¹ para Android e o Xcode² para iOS.
- **Gestão de Projeto (Issues/Tasks):** O software usado para a gestão do projeto foi o JIRA³. Este permite planear e organizar a construção de um produto bem como atribuir tarefas e seguir a atividade da equipa.
- **Gestão de Versões:** O sistema de gestão de versões utilizado foi o Subversion⁴ (mais conhecido por SVN). A principal justificação da utilização deste foi o requisito imposto pela empresa e equipa de trabalho, já que todo o material desenvolvido por esta, está e tem que ficar armazenado sob este sistema.

¹<http://www.eclipse.org/>

²<https://developer.apple.com/xcode/>

³<https://www.atlassian.com/software/jira>

⁴<http://subversion.apache.org/>

5.1 Processo de desenvolvimento

Para se criar uma aplicação pelo PhoneGap é preciso seguir as linhas orientadoras de como criar aplicações e implementá-las nos variados sistemas operativos móveis. Para se começar uma aplicação é preciso usar uma ferramenta *cordova* CLI, que permite criar novos projetos, compilar para as diferentes plataformas e executar a aplicação em dispositivos reais ou emuladores. Podemos usar o CLI para inicializar o código do projeto e após isto podemos usar os SDKs das diferentes plataformas para continuar o desenvolvimento.

5.1.1 Pré-requisitos

Antes de executar todas as ferramentas da linha de comandos é necessário instalar os SDKs de todas as plataformas para o qual pretendemos desenvolver.

Para adicionar uma nova plataforma ou reconstruir um projeto é preciso executar a interface de linha de comandos a partir da máquina que tem o suporte para aquele SDK, caso contrario é mostrado uma mensagem a indicar que a máquina não tem suporte para aquela plataforma.[Phoa]

O CLI suporta os seguintes plataformas:

- iOS
- Amazon Fogo OS
- Android
- BlackBerry 10
- Windows Phone 7
- Windows Phone 8
- Windows 8
- Firefox OS

5.1.2 Serviços da plataforma JOObian

Como referido anteriormente a aplicação móvel faz uso de uma série de serviços disponibilizados pela plataforma JOObian para disponibilizar a informação ao utilizador.

Foi utilizada a *framework* Swagger⁵ para descrever, consumir e visualizar os serviços REST que a plataforma disponibiliza.

Estes serviços foram divididos em quatro categorias:

- **Suporte:** Os serviços aqui disponibilizados não precisam de autenticação e disponibilizam informação como as indústrias, os países e as skills para o autocomplete.
- **Autenticação:** Nesta categoria temos os serviços de validação de registo, login e alteração de palavra passe.

⁵<https://helloreverb.com/developers/swagger>

- **Utilizador:** Estes serviços disponibilizam e guardam toda a informação que o utilizador consome, como as definições, o perfil, os favoritos entre outros.
- **Ofertas:** Serviço que disponibiliza as ofertas de emprego ao utilizador.

De seguida são detalhados todos os serviços utilizados na aplicação móvel JOObian.

Tipo	Nome	Tipo Retorno
GET	/industries	JSON
GET	/supportedCountries	JSON
GET	/skills	JSON

Tabela 5.1: Lista de serviços do suporte.

/industries: Serviço que disponibiliza todas as indústrias existentes na plataforma JOObian e que permitem definir em que indústria se enquadra o perfil do utilizador.

/supportedCountries: Serviço que disponibiliza todos os países existentes na plataforma JOObian e que permite ao utilizador definir quais os países de que deseja receber ofertas de emprego.

/skills: Este serviço é utilizado na definição de perfil do utilizador nomeadamente na criação de do perfil e edição das skills. para o autocomplete das skills. É passado um conjunto de letras e é devolvida uma lista com as skills que comecem por aquela letra ou letras. Este serviço é utilizado na definição de perfil do utilizador nomeadamente na criação do perfil e edição das skills.

Tipo	Nome	Tipo Retorno
GET	/validateRegister	JSON
POST	/register	JSON
POST	/login	JSON
POST	/credentials	JSON
PUT	/credentials	JSON

Tabela 5.2: Lista de serviços da autenticação.

/validateRegister: Serviço que verifica se um utilizador já existe ou não na plataforma JOObian. São passados os tokens do LinkedIn ou o correio eletrónico para validação. Este serviço é utilizado na fase inicial garantindo a unicidade dos registos.

/register: Serviço que regista um novo utilizador quer seja pelo LinkedIn ou através do registo JOObian. Para isso são passados os tokens do LinkedIn ou o correio eletrónico do utilizador e os dados do devido registo.

/login: Serviço que autentica o utilizador na plataforma JOObian e que devolve o token de sessão para os restantes serviços que necessitem de autenticação.

/credentials: Este serviço é utilizado nas duas perspetivas de alteração da palavra passe, quer seja através do formulário de alteração ou através do pedido de *reset* da palavra passe. Desta forma este serviço tem dois métodos, POST para alterar a palavra passe do utilizador

e PUT para *reset*, enviando uma nova palavra passe ou utilizador por correio eletrónico.

Tipo	Nome	Tipo Retorno
GET	/settings	JSON
POST	/settings	JSON
DELETE	/devices	JSON
POST	/devices	JSON
GET	/profile	JSON
GET	/favorites	JSON
POST	/favorites/{jobId}	JSON
DELETE	/favorites/{jobId}	JSON
POST	/skills	JSON
DELETE	/skills/{skillId}	JSON
POST	/statistics	JSON

Tabela 5.3: Lista de serviços do utilizador.

/settings: Serviço dividido em dois métodos. Método GET para devolver as definições do utilizador sempre que este entra na aplicação, pois este pode ter estado noutro dispositivo e alterado as suas definições. Método POST para guardar as mesmas do lado do servidor quando este altera alguma das suas definições.

/devices: Serviço dividido em dois métodos, DELETE para remover o dispositivo da plataforma JOObian para este não receber notificações da plataforma. POST para adicionar o dispositivo para receber notificações da plataforma JOObian. Este serviço é executado através da configuração de notificações na área de definições.

/profile: Serviço que disponibiliza a informação do utilizador. Este serviço é utilizado sempre que o utilizador acede à aplicação.

/favorites: Serviço que disponibilizar três métodos, GET que devolve todas as ofertas de emprego favoritas do utilizador. DELETE apaga a oferta de emprego dos favoritos do utilizador e POST adiciona uma oferta de emprego aos favoritos do utilizador na plataforma JOObian.

/skills: Serviço que disponibiliza dois métodos, POST adiciona uma nova skill ao perfil do utilizador e DELETE que elimina a skill do perfil do utilizador da plataforma JOObian.

/statistics: Serviço que disponibiliza a informação estatística do utilizador sobre as ofertas de emprego com base nas definições existentes no perfil e na ordenação pretendida pelo utilizador para apresentação de ofertas.

Tipo	Nome	Tipo Retorno
GET	/jobs	JSON

Tabela 5.4: Lista de serviços da oferta de emprego.

/jobs: Serviço que disponibiliza as ofertas de emprego. São passado como parâmetros a ordenação da lista, o índice da página e o número de ofertas de emprego que vão ser devolvidas.

5.1.3 Criação de uma aplicação em PhoneGap

A Figura 6.1 mostra o workflow do fluxo de criação de uma aplicação em PhoneGap.

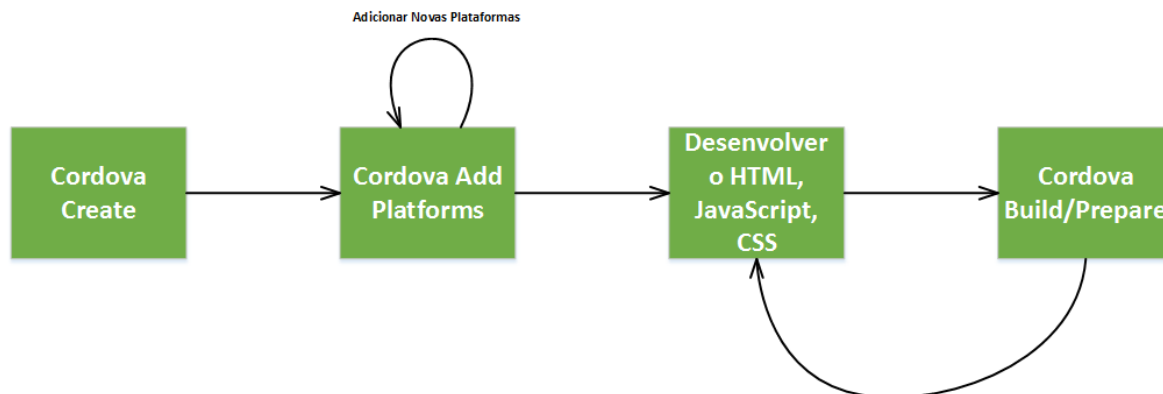


Figura 5.1: Workflow da criação de uma aplicação em PhoneGap.

De seguida são descritos cada um dos blocos presentes no workflow.

Cordova Create

A partir da interface da linha de comandos do *cordova* e estando dentro do diretório onde se quer manter o código de desenvolvimento e executa-se o seguinte comando para a criação de uma aplicação:

```
$ cordova create hello com.exemple.hello HelloWorld
```

O primeiro argumento *hello* especifica o nome do diretório onde todo o nosso projeto vai ser gerado. Se este diretório não existir é criado automaticamente. Dentro deste diretório é criada uma subdiretoria *www* onde se encontra a *home page* da aplicação, juntamente com outras diretorias *css*, *js* e *img*, que segue as convenções de nomeação de diretórios no desenvolvimento web. Gera também o ficheiro *config.xml* que contem meta-dados importantes e necessários para gerar e distribuir a aplicação.

O segundo argumento *com.exemplo.hello* aplica ao projeto um identificador de domínio. Este argumento é opcional, mas só se omitirmos o terceiro argumento. Podemos editar este valor mais tarde no ficheiro *config.xml*, tem-se que ter em conta que há código gerado fora do ficheiro *config.xml* como por exemplo o nome dos pacotes em Java. O identificador gerado por defeito é *io.cordova.hellocordova*, mas recomenda-se que se selecione um identificador de domínio com um nome apropriado ao projeto.

O terceiro argumento *HelloWorld* indica o título da aplicação. Este argumento também é opcional e pode ser editado mais tarde no ficheiro *config.xml*, tem-se que ter em conta que há código gerado fora do ficheiro *config.xml* como por exemplo o nome das classes do Java. O título gerado por defeito é *HelloWorld*, mas recomenda-se que se selecione um título apropriado para a aplicação.

Cordova Add Platforms

Todos os comandos subsequentes precisam de ser executados dentro do diretório do projeto, ou qualquer subdiretoria dentro do próprio.

Antes de poder construir o projeto é preciso especificar o conjunto de plataformas no qual a aplicação vai correr. Deve-se ter em conta que só é possível adicionar plataformas novas se o SDK da plataforma estiver instalado na máquina.

Para adicionar ou remover plataformas ao nosso projeto temos que executar os seguintes comandos:

```
$ cordova platform add android
$ cordova platform add ios
$ cordova platform add wp8

$ cordova platform remove blackberry10
$ cordova platform rm amazon-fireos
```

Quando é executado o comando de adição de uma plataforma é criada uma subdiretoria *platforms* e dentro desta são criadas tantas subdiretorias quantas as plataformas que foram adicionadas ao nosso projeto e gera um projeto específico para cada plataforma dentro do diretório *platforms*.

A partir deste ponto podemos usar os SDK's específicos de cada plataforma como alternativa para modificar e compilar o código que o *cordova* gerou dentro do diretório *platforms*.

Cordova Build

O comando a seguir é um atalho para os comandos *cordova prepare* e *cordova compile* que compila o código e deteta erros nos diversos projetos:

```
$ cordova build android
$ cordova build ios
$ cordova build wp8
```

O diretório *www* é copiado para cada uma das plataformas como por exemplo *platforms/ios/www* ou *platforms/android/assets/www*.

Todos os ficheiros HTML, JavaScript ou CSS3 devem ser adicionados, removidos ou alterados no diretório *www* e nunca na subdiretoria das *platforms*, porque a quando do processo de compilação todo o conteúdo destas é substituído pelo conteúdo do diretório raiz *www*. [Joh13]

O que foi apresentado até aqui é o modo como o *cordova* cria as aplicações e gera o código para cada uma das plataformas. Mas agora é preciso desenvolver toda a lógica da aplicação dentro do diretório *www* que consiste nas páginas HTML, no JavaScript e nos CSS.

5.2 Desenvolvimento Web

Considerando-se a perspetiva híbrida, recorreu-se as tecnologias típicas para este tipo de desenvolvimento como o HTML5, JavaScript e CSS3, AJAX e JSON.

O HTML5 foi especificado em 2007 e desde então têm vindo a crescer como padrão. O HTML5 tem sido trabalhado com o objetivo de dotar a Web com a capacidade de executar aplicações ricas, complexas e sofisticadas nos navegadores. Esta especificação trouxe consigo um conjunto de novas tecnologias, tais como novos elementos HTML, novas APIs, suporte a novas formas de multimédia, animações, CSS3, entre outras.[Dav10]

O CSS3 articula-se de forma próxima com o HTML5 sendo também a última versão do CSS. O CSS3 são um conjunto de instruções que definem o aspeto dos elementos HTML que resultam do conteúdo Web.[Dav12] Esta última versão do CSS3 trouxe consigo um novo conjunto de características que quando conjugadas com o HTML5 resultam em experiências ricas a todos os níveis. Algumas dessas características são os *Selectors* que permitem selecionar qualquer elemento a partir da relação entre os mesmos, as *media queries* que permitem definir diferentes características de um dado elemento de acordo com as características do dispositivo, por último as animações que permitem enriquecer a experiência de utilização.[Ma12]

Apesar de se conseguir produzir aplicações Web interessantes e visualmente apelativas recorrendo só ao HTML5 e ao CSS3, esta não tem qualquer capacidade para a implementação da lógica que está por detrás da aplicação. Para isso o JavaScript veio preencher essa lacuna, dotando os conteúdos Web com capacidade de processamento que lhes possibilita responder em tempo real ao utilizador. O JavaScript corre do lado do cliente o que permite efetuar alterações ao conteúdo sem necessitar de recarregar uma nova página.[Dav11]

Devido ao facto do JavaScript correr exclusivamente do lado do cliente acarreta uma limitação no processamento de dados cuja origem se encontra ou depende do lado do servidor. Para isso, o AJAX apresenta soluções, ao representar a interação entre diversas tecnologias. Através desta técnica é possível a uma aplicação fazer um pedido a um servidor, receber a resposta e apresentar de imediato a mesma sem ser necessário carregar/recarregar a página em questão.[Dav11]

Por fim, o JSON surgiu como um *standard* em 1999 e é sobretudo um formato simples de troca de informação entre cliente e servidor. Este é simples para os humanos lerem e escreverem e simples para as máquinas analisarem e gerarem. O JSON é um formato de texto que é completamente independente do idioma, mas usa convenções que são familiares aos programadores. Estas propriedades fazem do JSON uma linguagem de troca de informação ideal.[Ecm13]

As tecnologias referidas anteriormente enriquecem as potencialidades da Web, permitindo desenvolver produtos ricos e funcionais. Mas por si só estas tecnologias não são completas, pois é necessário implementar todas as funcionalidades de raiz como por exemplo, os mecanismos de navegação das aplicações, as animações das transições da aplicação, entre outros. No entanto, nos últimos anos têm surgido inúmeras ferramentas que facilitam o trabalho no desenvolvimento das aplicações tais como as *frameworks* (Capítulo 2, secção 2.3).

As opções no desenvolvimento desta aplicação recaíram sob o jQuery e sobre o AngularJS. O jQuery é uma biblioteca JavaScript, rápida, pequena e rica em funcionalidades. Estas características simplificam em muito a manipulação do HTML, o tratamento de eventos e simplifica o uso da API do AJAX. Com uma combinação de versatilidade e capacidade de expansão, o jQuery mudou a maneira como os *developers* programam o JavaScript.[The]

5.2.1 Configurar e utilizar o AngularJS

O código seguinte mostra um pequeno exemplo de um carrinho de compras que permite listar, adicionar e remover um produto do carrinho. Os dados serão armazenados num array de objetos JSON, que é um padrão excelente para trabalhar com o AngularJS. Vamos também ver como se deve utilizar o AngularJS na aplicação. Tendo por base este pequeno exemplo toda a aplicação JOObian nasceu a partir daqui.

```
<!DOCTYPE html>
<html ng-app>
  <head>
    <meta charset="utf-8">
    <title>AngularJS - Carrinho de Compras</title>
    <script src="lib/angular.min.js"></script>
  </head>
  <body ng-controller="HelloController">
    <h2>Carrinho de compras</h2>
    <ul>
      <li ng-repeat="item in carrinho">
        {{ item.nome }}
        <a href="" ng-click="removeItemCarrinho($index)">remove
        item</a>
      </li>
    </ul>
    <form ng-submit="adicionarItemCarrinho()">
      <input type="text" ng-model="itemAdicionar" />
      <input type="submit" value="Adicionar ao carrinho" />
    </form>
    <script>
      function HelloController( $scope ) {

        $scope.carrinho = [ { nome : 'Livro AngularJS' }, {
          nome : 'Livro Aprenda AngularJS em 24h' } ];

        $scope.adicionarItemCarrinho = function() {
          $scope.carrinho.push( { nome :
            $scope.itemAdicionar } );
          $scope.itemAdicionar = '';
        };

        $scope.removeItemCarrinho = function( index ) {
          $scope.carrinho.splice( index, 1 );
        };

      }
    </script>
  </body>
</html>
```

A única configuração necessária para poder usar AngularJS na nossa aplicação é importar o ficheiro “angular.min.js” na nossa página como mostra o exemplo a seguir.

```
<script src="lib/angular.min.js"></script>
```

Como se pode perceber a maioria do código é basicamente HTML há exceção dos recursos do AngularJS, que são as diretivas `ng-app`, `ng-controller`, `ng-repeat`, `ng-click` e `ng-submit`.

A diretiva `ng-app` é responsável por inicializar uma aplicação AngularJS. Normalmente ela é usada nas tags de maior desígnio como a tag *html* ou *body*.

O operador `{{...}}` é equivalente à diretiva `ng-bind` e serve para exibir o conteúdo de algum item contido no contexto da aplicação, como por exemplo o valor atribuído a um item no `ng-repeat`.

A diretiva `ng-controller` é responsável por associar um controle na view da aplicação. Isto é, determinado elemento do HTML vai ser controlado por código onde a `ng-controller` foi definida neste caso temos a função `HelloController` dentro da tag *script* que representa a nossa classe e que foi associada à diretiva `ng-controller`.

A diretiva `ng-repeat` permite iterar e mostrar os dados de um array na nossa view. Todos os itens iterados é disponibilizada a variável `$index`, que contém o índice o item no array.

A diretiva `ng-click` como o próprio nome indica é a ação de clique sobre um objeto, como um link ou botão. Para que esta diretiva funcione é preciso passar uma função contida dentro do controle da view.

A diretiva `ng-submit` é igual ao `ng-click`, porém é utilizada para envio e processamento de formulários. Também evita que o formulário seja efetivamente enviado ao clicar no botão de submit, prevenindo assim que a página fosse recarregada.

A função `HelloController` recebe um parâmetro como argumento, o `$scope` que é o contexto do nosso controlador, isto é, tudo o que está na hierarquia onde definimos a nossa diretiva `ng-controller` essa variável `$scope` é passada por injeção de dependências pelo próprio AngularJS.

Todos os atributos e funções que estão associados à variável `$scope` no controlador, significa que estas podem ser acedidas na view, assim quando declaramos a diretiva `ng-repeat` com a variável “carrinho” o AngularJS sabe onde encontrar os dados a ser mostrados. O mesmo acontece para as funções utilizadas no `ng-click` e `ng-submit`, responsáveis por processarem as ações da página.

5.2.2 Aplicação JOObian

Para ajudar a compreender o desenvolvimento realizado a Figura 6.2 representa um diagrama de contextualizao dos principais grupos de funcionalidades implementadas na aplicação móvel JOObian. Cada uma destas funcionalidades vai ser exposta recorrendo a um *storyboard* onde são explicados sucintamente cada um dos ecrãs que o compõem.

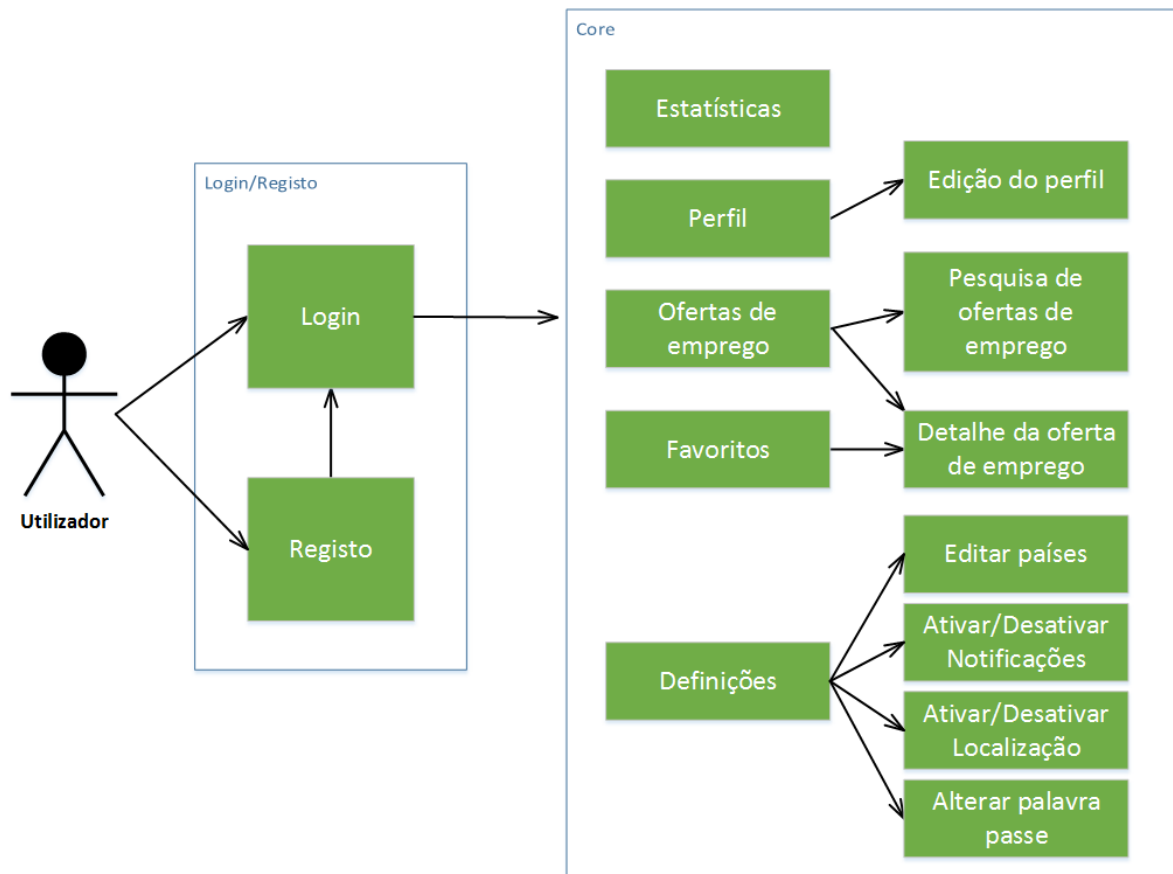


Figura 5.2: Diagrama de funcionalidades da aplicação móvel JOObian.

5.2.3 Login

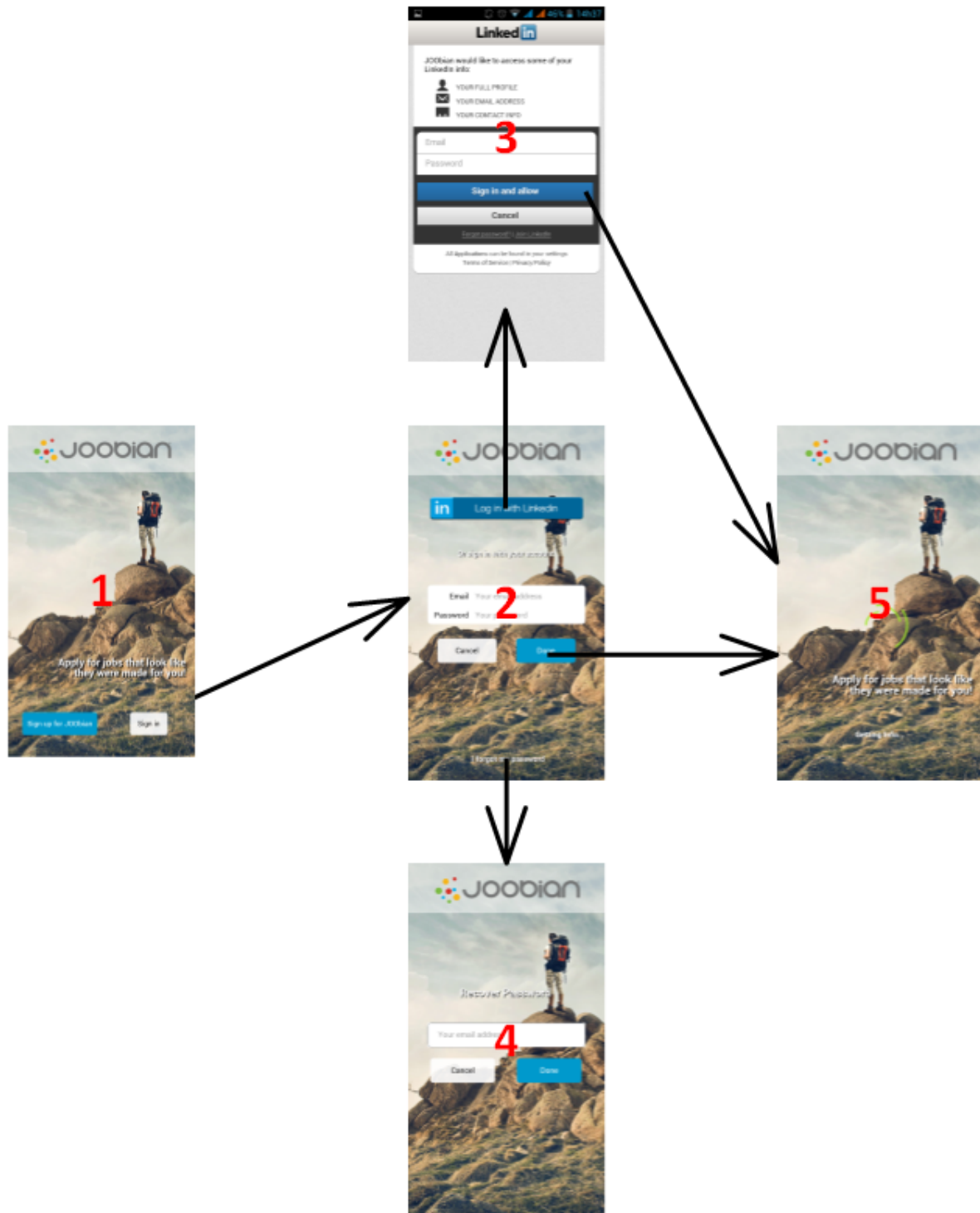


Figura 5.3: *Storyboard* de login da aplicação móvel JOObian.

Ao entrar na aplicação móvel JOObian (ecrã 1) é verificado se existe conectividade à Internet, uma vez que aplicação precisa de comunicar com o servidor. Se não houver conectividade à Internet é mostrada uma mensagem ao utilizador a indicar o mesmo e de 5 em 5 segundos é feita uma nova tentativa para averiguar se já existe conectividade. Assim que houver ligação à Internet os botões de *Sign up for JOObian* e *Sign in* aparecem no ecrã e a partir deste ponto o utilizador pode efetuar o registo na plataforma JOObian caso não tenha conta ou então efetuar login na aplicação.

Um utilizador com o processo de login feito anteriormente não precisa de voltar a inserir os seus dados ao entrar na aplicação, esta automaticamente entra no core da aplicação.

O diagrama da Figura 6.4 mostra a lógica da aplicação no momento em que o utilizador entra nesta.

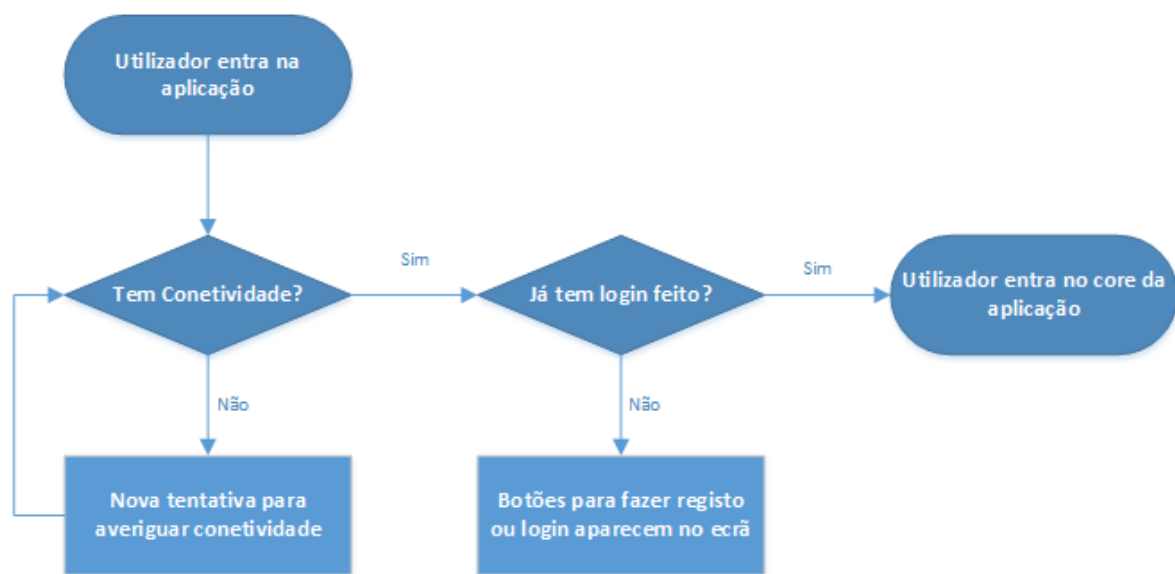


Figura 5.4: Diagrama de fluxo do ecrã inicial da aplicação móvel JOObian.

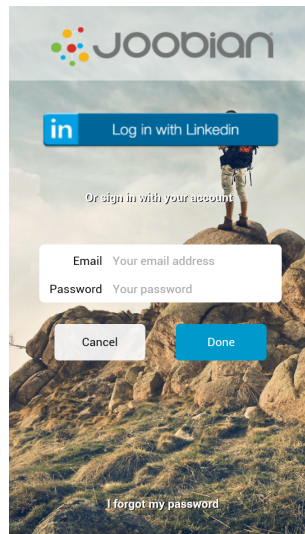


Figura 5.5: Ecrã de “Sign in” da aplicação móvel JOObian.

O ecrã 2 representa os dois métodos que o utilizador tem para fazer login na plataforma JOObian. Um utilizador já registado no sistema JOObian tem a possibilidade de fazer login via LinkedIn caso tenha escolhido esse processo na fase de registo ou então pela inserção manual dos seus dados para isso introduzindo o seu endereço de correio eletrónico e palavra passe. Em ambos os processos caso a conta não exista o utilizador é encaminhado para o processo de registo.

Se o utilizador escolher o LinkedIn este é encaminhado para o ecrã 3 (Figura 6.6) para inserir os seus dados (correio eletrónico e palavra passe) da sua conta de LinkedIn.

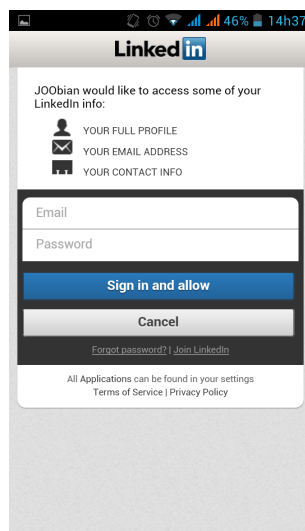


Figura 5.6: Ecrã da página de login do LinkedIn.

Neste processo de login pelo LinkedIn é necessário usar a API dos mesmos para que seja mostrado o ecrã 3 (Figura 6.6). Durante todo o processo de comunicação com a API do

LinkedIn é usado OAuth que é um protocolo que permite autorizações seguras para obter acesso a serviços HTTP.

São então enviadas pelos serviços do LinkedIn as *apiKeys* atribuídas ao JOObian. Se tudo estiver correto os serviços do LinkedIn retornam dois tokens (público e privado) e com a duração de 599 segundos (10 minutos).

É então feito um novo pedido aos serviços do LinkedIn com o token público enviado anteriormente para que seja mostrada a janela de login.

Após o utilizador inserir corretamente os seus dados os serviços do LinkedIn retornam mais dois tokens (público e privado), neste caso são os tokens que identificam o utilizador e é com estes mesmos que é efetuado o pedido da informação do utilizador ao LinkedIn.

O diagrama da Figura 6.7 mostra a sequencia e métodos utilizados no processo de autenticação do LinkedIn.

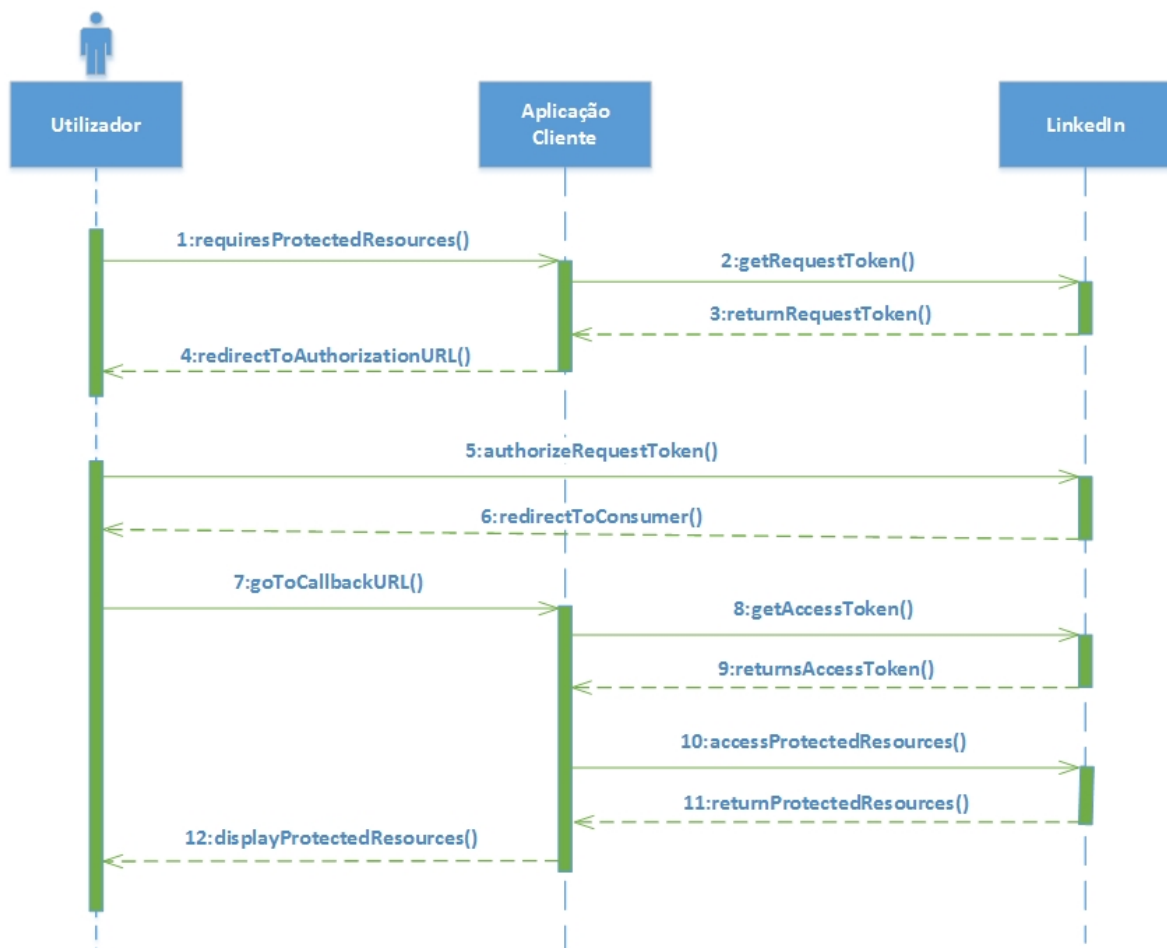


Figura 5.7: Diagrama de sequência do protocolo OAuth.

Após receber estes tokens do LinkedIn são enviados ao servidor para averiguar se o utilizador já existe na plataforma JOObian. Se a resposta for positiva é enviado ao servidor um pedido de login na plataforma Joobian no qual este responde com um token de sessão que tem

duração de uma hora. Este token é utilizado em todos os pedidos de informação ao servidor daquele utilizador.

Para ajudar a compreender este processo o diagrama da Figura 6.8 mostra todo o processo que existe na autenticação pelo LinkedIn e posterior autenticação na plataforma JOObian.

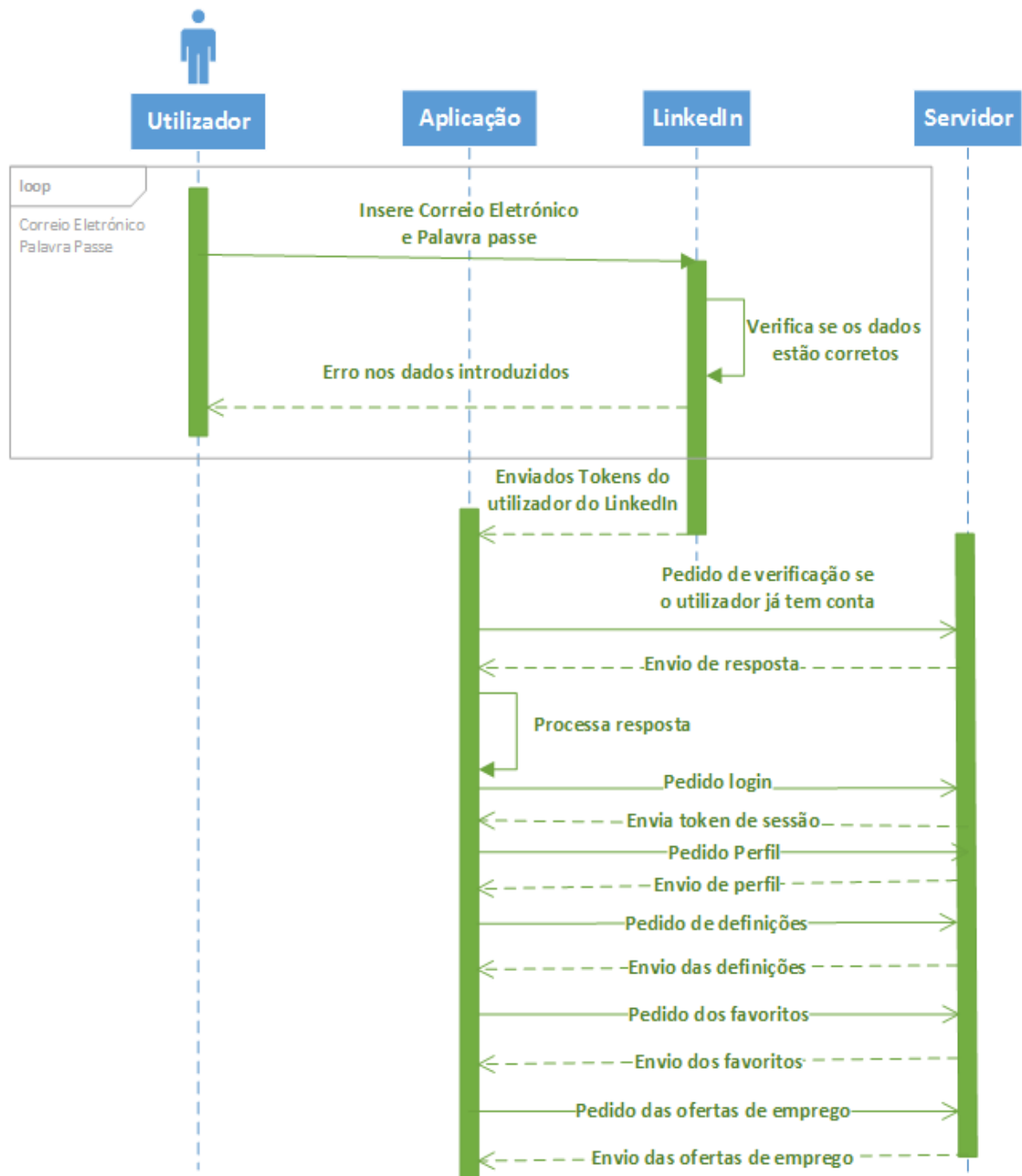


Figura 5.8: Diagrama de sequência no processo de login pelo LinkedIn.

Se o utilizador escolher o processo manual, os campos de *input* são validados assim que este carrega no botão de “Done”. O correio eletrónico passa por uma expressão regular que valida se é um correio eletrónico bem formado.

Exemplo da expressão regular aplicada ao endereço de correio eletrónico:

```
var reg = /^[^<>() [\]\.\,;\s@"]+(\.[^<>() [\]\.\,;\s@"]+)*|(\".+\")@((\[[0-9]{1,3}\. [0-9]{1,3}\. [0-9]{1,3}\. [0-9]{1,3}\)|((\a-zA-Z\d-0-9)+\.)+[a-zA-Z]{2,}))$/;
```

A palavra passe tem que ter no mínimo seis caracteres. Se a palavra passe introduzida pelo utilizador estiver incorreta é mostrado uma mensagem de aviso ao utilizador que as credenciais estão incorretas.

Após todas as validações o processo de login é igual ao descrito no processo do LinkedIn. Por questões de segurança a palavra passe do utilizador enviada no processo de login é cifrada usando o algoritmo MD5.

O AngularJS disponibiliza uma biblioteca JavaScript “Local Storage” que permite armazenar conteúdo, que o torna acessível em todo o contexto da aplicação móvel.

A “Local Storage” é usada no processo de login para guardar os tokens do utilizador do LinkedIn ou então para guardar o endereço de correio eletrónico e a palavra passe cifrada.

De seguida é dado um exemplo de como se manipula os dados na “Local Storage”:

- **Adicionar à local storage:** `localStorageService.add('localStorageKey','Add this!');`
- **Ler um valor da local storage:** `var value = localStorageService.get('localStorageKey');`
- **Remover um valor da local storage:** `localStorageService.remove('localStorageKey');`
- **Remover toda a local storage:** `localStorageService.clearAll();`
- **Criar cookies usando os mesmos métodos anteriores:**
`localStorageService.cookie.add('localStorageKey','I am a cookie value now');`

O utilizador é então encaminhado para um processo de espera (ecrã 5) e onde é mostrado o que a aplicação está a fazer em *background*. Durante este processo a aplicação móvel pede o perfil, as definições, os favoritos e as ofertas de emprego do utilizador que acabou de fazer login. Todo este processo completo o utilizador tem acesso as ofertas de emprego e aos seus dados.

No fundo do ecrã 2 encontra-se um link para o utilizador recuperar a sua palavra passe. O utilizador é encaminhado para o ecrã 4 onde é pedido para inserir o endereço de correio eletrónico utilizado na plataforma JOObian. Quando o utilizador carrega no botão de “Done” o endereço de correio eletrónico passa de novo pela expressão regular referida acima para averiguar se esta bem formado. Após isto é enviado um “email” para o correio eletrónico do utilizador com a sua nova palavra passe, esta pode ser alterada mais tarde dentro da aplicação móvel.

5.2.4 Registo

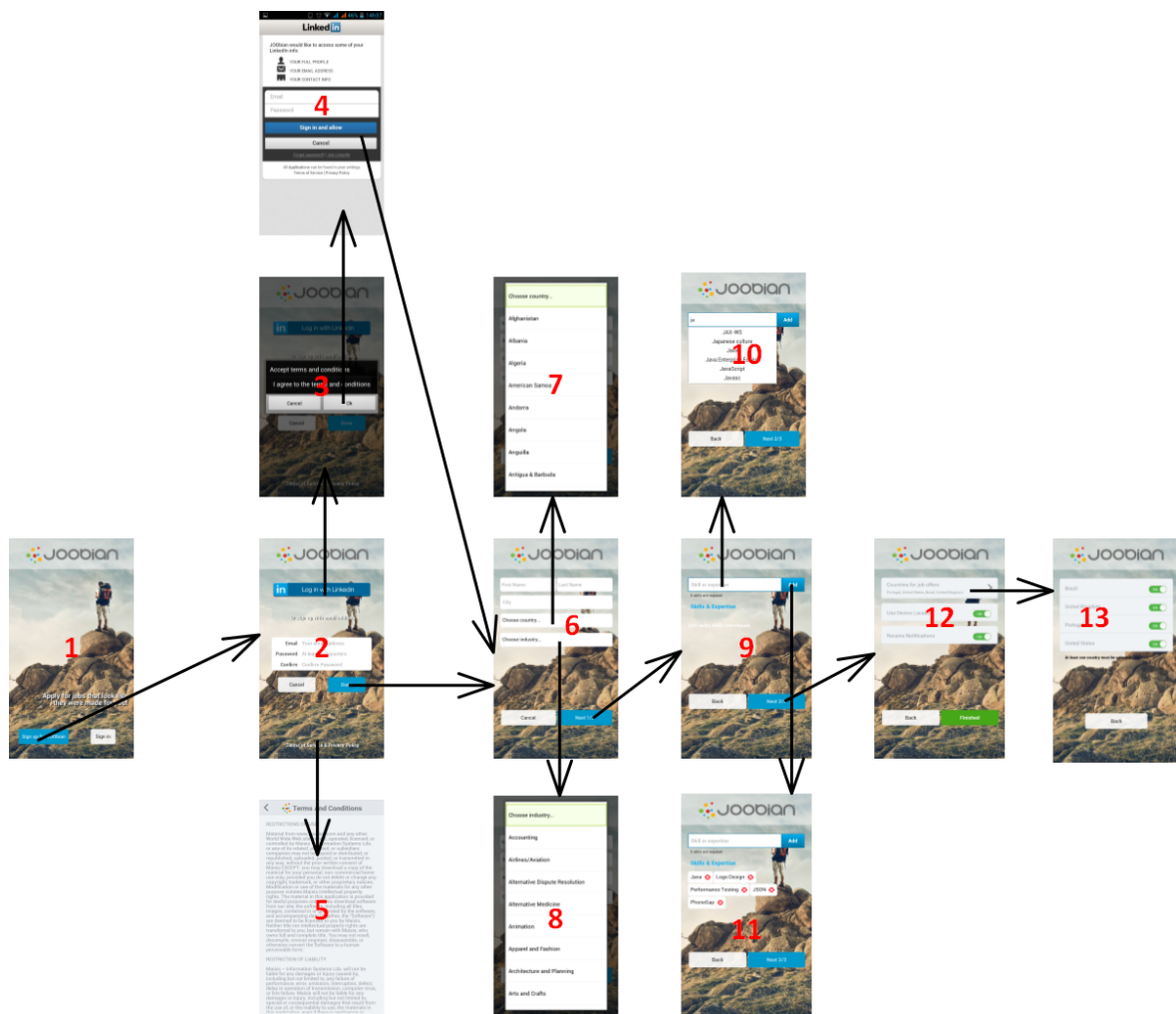


Figura 5.9: *Storyboard* do registo da aplicação móvel JOObian.

Quando o utilizador entra na aplicação é verificado se esta tem conetividade à Internet, tal como o processo já foi descrito na secção de Login. Sendo agora abordado os outros pontos do *storyboard*.

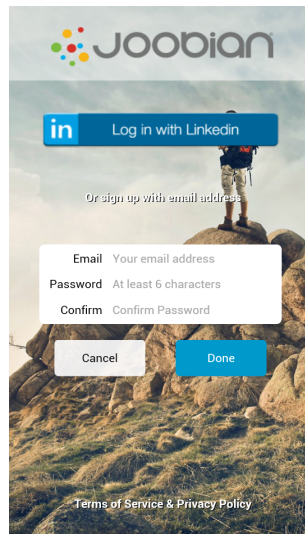


Figura 5.10: Ecrã 2 de “Sign up” da aplicação móvel JOObian.

Após o utilizador carregar no botão de *Sign up for JOObian* no ecrã 1 o utilizador é encaminhado para o ecrã 2 onde este pode escolher entre os dois processos de registo, via LinkedIn ou inserindo manualmente os seus dados.

É também mostrado no fundo do ecrã 2 um link para os termos de serviços e política de privacidade da aplicação móvel JOObian. O utilizador ao carregar neste link é encaminhado para o ecrã 5 onde pode ler os mesmos.

Se o utilizador escolher o LinkedIn é-lhe mostrado um aviso com uma pergunta se aceita os termos e política de privacidade da aplicação (ecrã 3), se o utilizador aceitar os termos este é encaminhado para o ecrã 4, processo de login pelo LinkedIn (processo descrito na secção de Login)

Após o processo de login ser feito com sucesso na plataforma do LinkedIn é feita uma verificação na plataforma JOObian se aquela conta ainda não existe. Se existir é feito login automaticamente no nosso sistema e o utilizador acede aos dados.

Para ajudar a compreender este processo o diagrama da Figura 6.10 descreve esse processo.

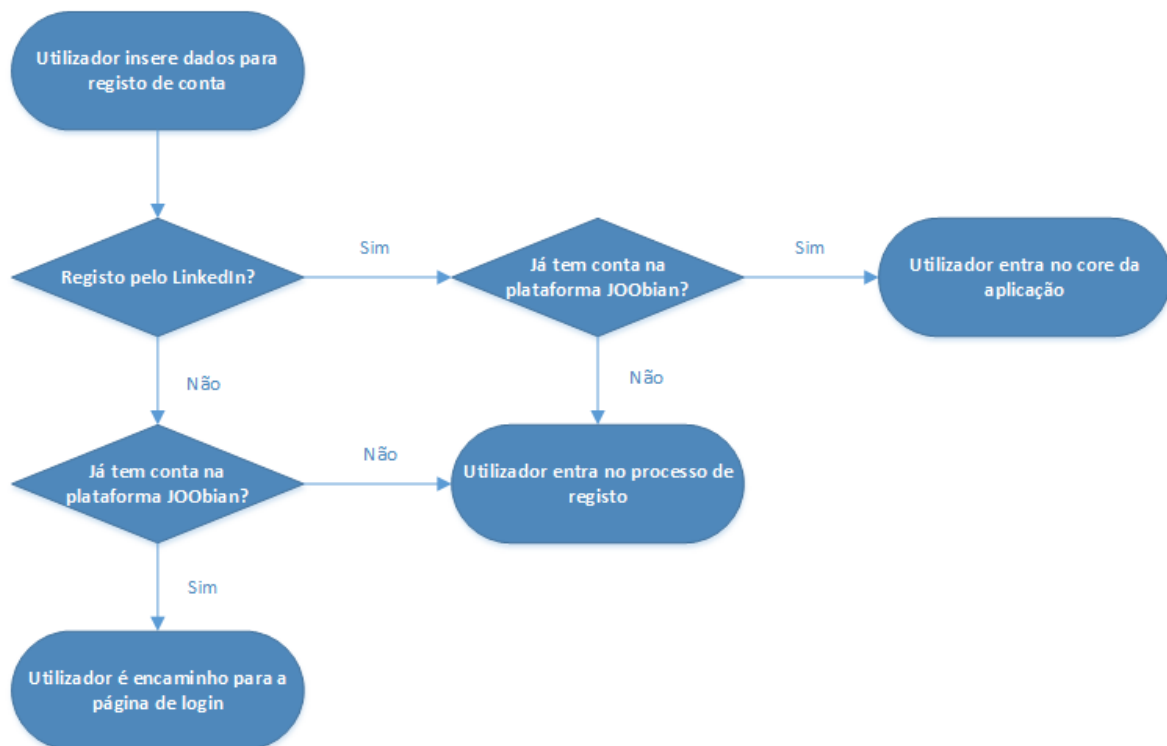


Figura 5.11: Diagrama de fluxo no registo de conta na plataforma JOObian.

Como estamos então no processo de registo a informação pedida ao LinkedIn do utilizador é:

- Correio eletrónico
- Primeiro e último nome
- Cidade e país
- Skills
- Indústria

O pedido é feito da seguinte maneira:

```

linkedinService.OAuth1GetInfo('first-name,last-name,skills,industry,email-address,
location:(name)', localStorageService.get("oauthUserToken"),
localStorageService.get("oauthUserTokenSecret"))
  
```

Para se obter esta informação, são sempre passado os tokens do utilizador já previamente guardados na local storage no processo de login. A informação recolhida é guardada em local storage para os passos seguinte.

Se o utilizador optar por colocar os dados manualmente no processo de registo é então solicitado ao utilizador que insira um correio eletrónico, uma palavra passe e a confirmação da palavra passe.

Assim que o utilizador carregar no botão de “Done” os campos são validados. O correio eletrónico passa pela expressão regular referida no processo de Login que valida se é um correio eletrónico bem formado.

Para dar alguma segurança à palavra passe são pedidos no mínimo 6 carateres, se esta não atingir este número é mostrado ao utilizador um aviso.

Exemplo de uma mensagem de aviso ao utilizador:

```
if($('#pass').val().length < 6)
    navigator.notification.alert(
        "Your password has not 6 characters!",
        callBackFunctionB, // Specify a function to be called
        'Alert',
        "OK"
    );
```

A confirmação da palavra passe serve para garantir que o utilizador não se enganou a inserir a palavra passe. Caso estas não coincidam é mostrada um aviso ao utilizador do mesmo género do exemplo anterior.

Estes campos respeitam as boas práticas na programação em HTML. O *input* do correio eletrónico é do tipo *email* o que faz com que o teclado no dispositivo móvel apresente os carateres necessários à inserção de um campo deste género.

E os *inputs* da palavra passe e confirmação da mesma são do tipo *password* que faz com que não seja mostrada a palavra passe em aberto substituindo-a por pontos ou asteriscos. A palavra passe é cifrada usando o algoritmo MD5 e guardada para mais tarde ser enviada no processo final do registo.

Uma vez mais verifica-se se o endereço de correio eletrónico existe na plataforma JOObian, em resposta afirmativa o utilizador é encaminhado para a página de *Sign in*.

Figura 5.12: Ecrã 6 da primeira página da fase de registo.

Após o processo anterior estar todo validado é apresentado o ecrã 6 (Figura 6.11), onde é pedido ao utilizador para inserir o seu primeiro e último nome, a sua cidade e país e a sua área de estudo (Indústria). Caso o utilizador tenha feito o registo a partir do LinkedIn toda esta informação já vai estar preenchida tendo este a possibilidade de a editar.

De referir que os campos “country” e “industry” são duas “select box” de onde o utilizador pode escolher um país (ecrã 7) e uma área de estudo (Indústria; ecrã 8) respetivamente. A “select box” “country” é preenchida com dados estáticos definidos no HTML. Na “select box” “industry” é usado um serviço para ir buscar todas as indústrias ao sistema e populada com a informação recebida do servidor.

Exemplo do serviço para ir buscar as indústrias ao sistema:

```
$.get(global_getSupportIndustries, function(data) {
    $rootScope.industries = data;
});
```

Resposta enviada pelo servidor:

```
{
  "industries": ["Accounting:47", "Airlines/Aviation:94", "Alternative Dispute
    Resolution:120", "Alternative Medicine:125", "Animation:127", "Apparel and
    Fashion:19", "Architecture and Planning:50", "Arts and
    Crafts:111", "Automotive:53", ..., "Wine and
    Spirits:142", "Wireless:119", "Writing and Editing:103"]
}
```

Todos os campos são de preenchimento obrigatório e o utilizador só avança no processo assim que todos tiverem preenchidos.

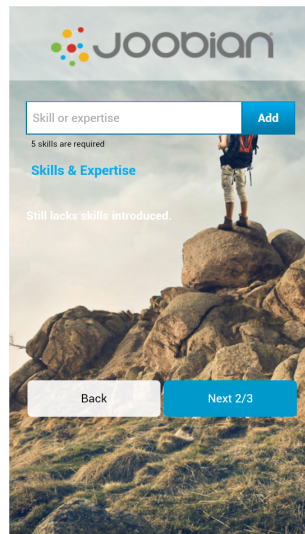


Figura 5.13: Ecrã 9, segundo passo da fase de registo.

Nesta ecrã o utilizador insere as suas habilitações (skills) da sua área de estudo (indústria). O campo onde o utilizador insere o nome da habilitação ou experiência usa “autocomplete” para ajudar o utilizador na processo de escolha (ecrã 10). Se a skill ou experiência não existir no nosso sistema o utilizador pode adiciona-la à mesma ao seu perfil, ficando num estado de aprovação na plataforma JOOBian. O sistema fica desta forma protegido de utilizadores mal intencionados e que não geram conhecimento para a plataforma.

Uma vez mais, se o utilizador usou o LinkedIn no processo de registo e este tiver habilitações no mesmo, estas já vão estar populadas na página e pode adicionar ou remover mais habilitações.

São obrigatórias no mínimo cinco skills e um máximo de cinquenta, foi imposto este limite mínimo para dar resultados mais fiáveis ao utilizador, o limite máximo é o mesmo do LinkedIn para não devolver ofertas de emprego já fora do seu âmbito.

O “autocomplete” é uma diretiva usado no campo de *input* que foi adaptada para consumir o serviço do sistema que devolve uma lista de até seis skills conforme aquilo que o utilizador estiver a escrever.

Exemplo do serviço que popula o “autocomplete” com a informação:

```
$.ajax({
  url: global_getSupportskills + "?text=" + encodeURIComponent(request.term) +
    "&maxResults=6",
  dataType: "json",
  success: function(data) {
    response(data.results);
  }
});
```

Como a skill ou experiência é passada nos argumentos da URL do serviço, utiliza-se a função JavaScript `encodeURIComponent()` para codificar o skill ou experiência devido aos caracteres especiais que estas possam ter (e.g. C++, C#).

O diagrama da Figura 6.13 ilustra todo o processo de comunicação entre o cliente e o servidor.

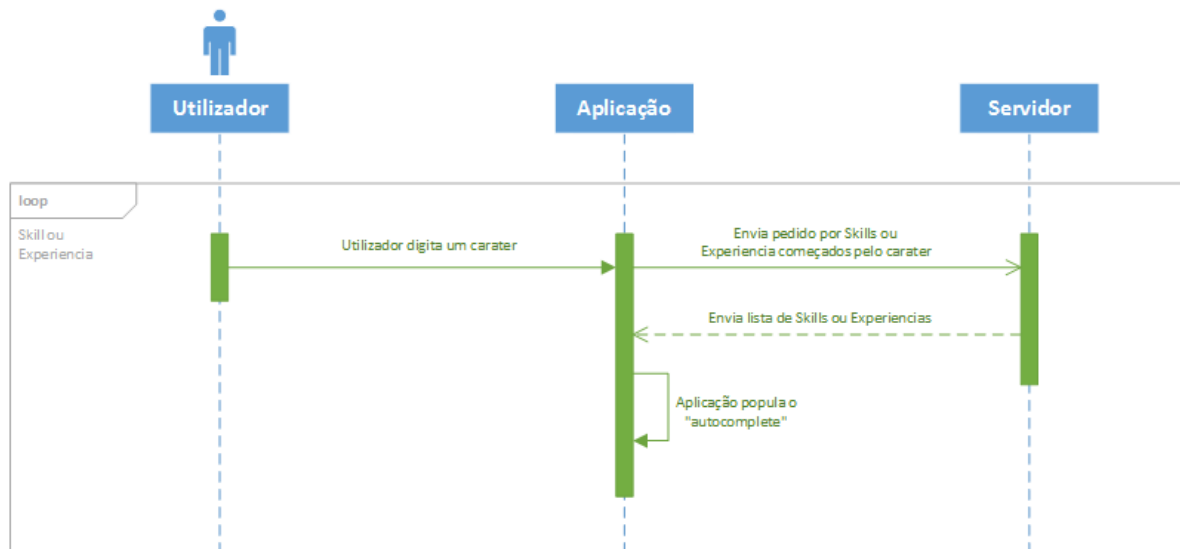


Figura 5.14: Diagrama de sequencia do processo de comunicação entre o cliente e o servidor.

Este serviço tem que devolver o resultado o mais rápido possível para que o utilizador não seja obrigado a escrever todo o texto da skill ou experiência, poupando assim algum tempo neste processo de escolha de skills.

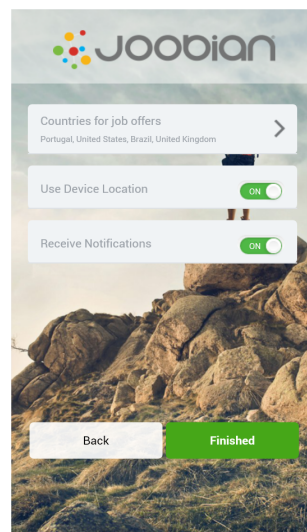


Figura 5.15: Ecrã 12 da terceira fase de registo.

Nesta ultima fase do processo de registo na plataforma JOObian (ecrã 12, Figura 6.14) é mostrado ao utilizador todos os países selecionados do total de países existentes no sistema,

tendo a possibilidade de remover e voltar a adicionar aqueles que forem da sua preferência (ecrã 13). Pode ainda desativar de receber notificações e se permite usar a localização do dispositivo.

Todos estes dados do processo de registo são guardados num objeto JSON que é enviados para o sistema assim que o utilizador carregar no botão de “Finished”. É efetuado o registo do utilizador e o sistema começa logo à procura das melhores ofertas de emprego para o utilizador.

Exemplo do objeto em JSON enviado ao servidor para registo do utilizador:

```
"{"  
  "joobianClientInfo":{"  
    "email":"teste@maisis.pt",  
    "firstName":"Teste",  
    "lastName":"Maisis",  
  "location":{"  
    "country":"Portugal",  
    "region":"Aveiro"  
  }},  
  "joobianClientExpertise":{"  
    "industry":"Computer Software",  
    "skills":["Java","Android Development",".Net","C#","C/C++","SQL"]  
  },  
  "joobianClientSettings":{"  
    "countries":["PT","US","BR","GB","IE"],  
    "email":"false",  
    "notification":"false",  
    "geoLocation":"true"  
  }  
}"
```

Conclusão

As principais dificuldades na implementação do registo prendeu-se com o desenvolvimento das páginas HTML, pois para cada resolução de dispositivo o layout ficava todo desformatado mesmo usando percentagens nos seus estilos. A solução para este problema foi criar “media queries” no CSS3 para cada resolução de dispositivo mais utilizado no mercado.

A integração com o LinkedIn foi outro ponto crítico onde foi despendido bastante tempo na implementação do mesmo, pois foi preciso saber todos os processos de comunicação com os serviços de autenticação do LinkedIn. Outro ponto teve haver com a implementação do “autocomplete” no processo de inserção das skills. A diretiva estava implementada para usar informação estática e foi necessário que perceber o mecanismo de funcionamento da mesma para poder usar informação dinâmica com a utilização dos serviços do sistema.

5.2.5 Pilha de Ofertas de Emprego

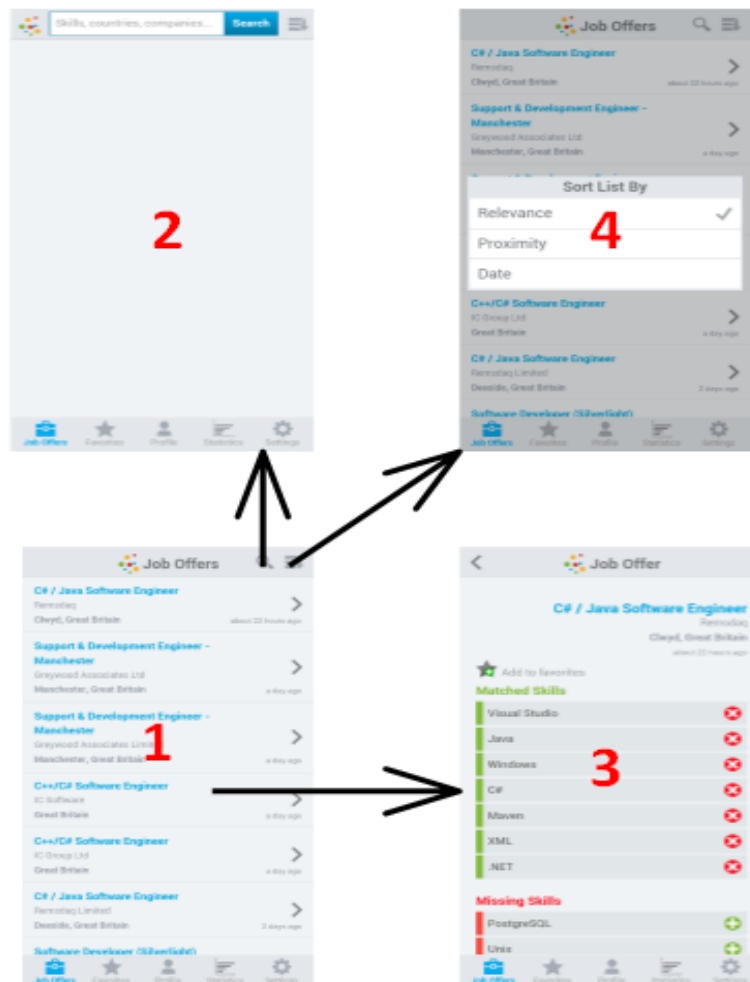


Figura 5.16: *Storyboard* da pilha de ofertas de emprego.

A partir deste ponto o utilizador tem acesso ao core da aplicação móvel JOObian. Pode ver as ofertas de emprego recomendadas, os seus favoritos, o seu perfil, as estatísticas do seu perfil sobre as ofertas de emprego e ver as definições presentes na aplicação.

No ecrã 1 para além de poder ver a pilha de ofertas de emprego recomendadas tem também algumas *features* interessantes. No primeiro pedido feito ao servidor por ofertas de emprego, que acontece quando o utilizador está a entrar na aplicação móvel, este devolve uma lista de vinte ofertas de emprego. A primeira *feature* implementada uma diretiva na lista das ofertas de emprego, quando o utilizador começa a navegar na lista e chega ao fim desta é despoletado um evento que vai pedir mais vinte ofertas de emprego ao servidor num máximo de cem.

Outras das *features* implementadas é a diretiva “pull to refresh”, isto é, se o utilizador estiver no topo da lista e puxar esta para baixo é despoletado um evento que vai pedir ao servidor as novas ofertas de emprego que entretanto tenham chegado ao sistema e que se encontrem dentro das recomendações para o utilizador.

O utilizador tem ainda neste ecrã a possibilidade de ordenar a pilha de resultados e esta pode ser ordenada dos seguintes modos (ecrã 4):

- **Relevância:** Os resultados apresentados são as ofertas de emprego que tem mais correspondência de skills entre a oferta e o utilizador.
- **Proximidade:** Se o utilizador tiver a localização ativa no dispositivo móvel, os resultados apresentados são as ofertas de emprego que estão mais próximas da sua localização e que maior relevância tem para o seu perfil. Caso a localização não esteja ativa, são apresentadas as ofertas de emprego próximas da cidade que o utilizador tem no seu perfil.
- **Data:** Os resultados apresentados são as ofertas de emprego com maior relevância ordenadas por data da mais atual para as mais antigas.

Foi implementado uma diretiva na data da oferta de emprego de forma a tornar perceptível ao utilizador a compreensão da mesma. Do servidor vem a data no formato “2014-04-27T06:21:01” e foi alterada para indicar a quantas horas, dias ou meses a oferta foi publicada, como por exemplo “about 5 hours ago”, “a day ago”, “a month ago”.

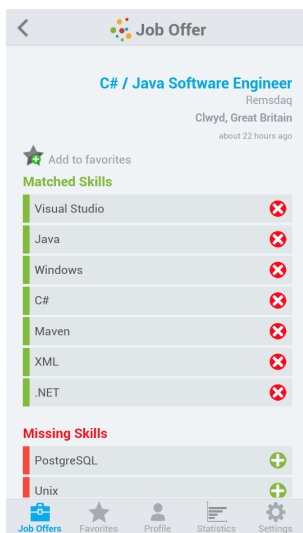


Figura 5.17: Ecrã 3 com o detalhe da ofertas de emprego.

No ecrã 3 o utilizador fica com uma ideia global do seu perfil com a oferta de emprego, são apresentadas as skills que fizeram correspondência com as da oferta e as que estão em falta ao perfil do utilizador.

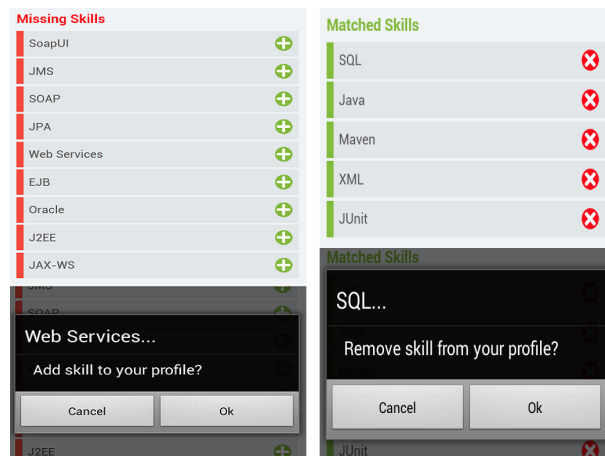


Figura 5.18: Adicionar ou remover skill do perfil.

O utilizador ao carregar nos botões que se encontram no fim de cada elemento da lista tem a possibilidade de adicionar ou remover skills do seu perfil. Quando este adiciona ou remove uma skill é guardada na localstorage uma variável a indicar que o perfil do utilizador mudou e a nova informação é enviada ao servidor e atualizado o perfil do mesmo na plataforma JOObian. Quando o utilizador volta ao ecrã 1 é feito um pedido de ofertas de emprego ao servidor tendo em conta as alterações que houve no perfil.

Por vezes o texto da descrição da oferta de emprego tem um tamanho considerável e torna-se difícil perceber onde é que estão as skills no meio do texto. Para ajudar o utilizador a descobrir as skills no meio do texto foi implementado um mecanismo de *highlight* para isso o utilizador só tem que carregar no texto da pilha de skills. Se carregar numa “match skill” é feito o *highlight* a verde caso contrario o *highlight* é feito a vermelho.

Experiência mínima de 2 anos em desenvolvimento **Java** J2EE ; Desenvolvimento de **Web Services** de SOAP **XML** EJB JAXWS JAXB SQL JPA JMS Maven Ferramentas de teste : SOAPUI JUnit Base de dados : **Oracle** Postgres SQL préférentiel ; Idioma : Français falado e escrito muito valorizado e Inglês Obrigatório Funções a desempenhar: Participação no projecto de arquitectura ; Participação na conceção dos ambientes de testes e validação ; Elaboração de serviços de IST ; Codificação e teste de unidade de módulos ; Integração do sistema técnico ;

Figura 5.19: *Highlight* das skills nas ofertas de emprego.

Quando o utilizador entre no ecrã 3 é corrida a lista de ofertas de emprego dos favoritos do utilizador, indicando assim se a oferta que ele esta a ver é ou não uma oferta dos seus

favoritos. Se não for, o utilizador tem a possibilidade de adicionar a oferta aos seus favoritos para mais tarde poder voltar a consultar a oferta de emprego.

No fim do detalhe todo da oferta de emprego encontra-se um botão “Apply on Indeed” que encaminha o utilizador para a oferta de emprego no contexto do provedor da oferta.

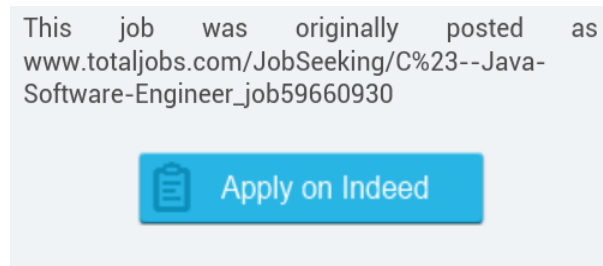


Figura 5.20: Botão para ver oferta de emprego no provedor da oferta de emprego.

No ecrã 1 o utilizador tem ainda a possibilidade de poder fazer pesquisas, para isso basta carregar na imagem da lupa que se encontra no canto superior direito do ecrã e é então encaminhado para o ecrã 2. Neste o utilizador pode pesquisar ofertas de emprego por skills, cidade, país, indústria ou empresa. A pilha de resultado retornada vai estar dividida em dois, isto é, a primeira pilha de ofertas de emprego são aquelas que fazem correspondência com o perfil do utilizador; a segunda pilha são todas as ofertas de emprego quer sejam da sua área de estudo ou não.

O utilizador também pode ordenar estas pilhas de resultados por relevância, proximidade ou data.

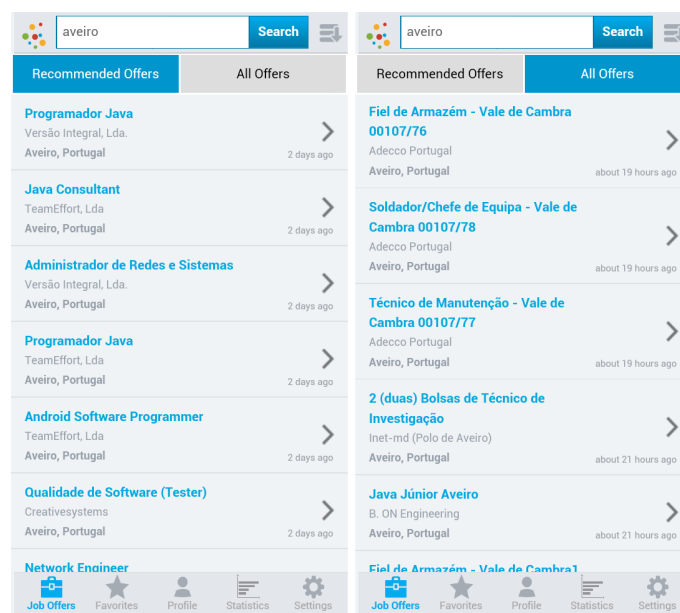


Figura 5.21: Ecrã da pesquisa com as duas pilhas de ofertas de emprego.

5.2.6 Favoritos

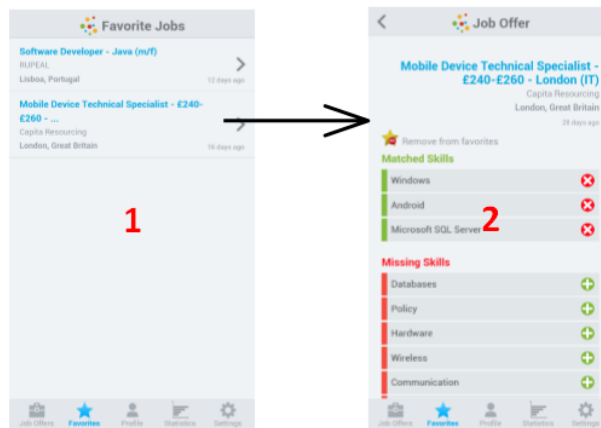


Figura 5.22: *Storyboard* dos favoritos do utilizador.

O utilizador ao entrar nos favoritos (ecrã 1) é mostrada a sua pilha de ofertas de emprego favoritas ordenadas por data de inserção. Os favoritos já foram pedidos no processo de login da aplicação. Esta ordenação é feita quando a pilha esta a ser carregada na view, através da diretiva “ng-repeat” utilizando o filtro “orderBy” passando o atributo pelo qual a lista vai ser organizada.

Exemplo da ordenação dos favoritos.

```
ng-repeat="favoriteJob in favoriteJobs.jobOffers.jobOffersResult |  
  orderBy: '-favoritedDate' "
```

Nesta lista o utilizador pode navegar para dentro da oferta favorita (ecrã 2) e fazer todos os processos descritos na secção Pilha de ofertas de emprego (Detalhe da oferta de emprego).

Sempre que o utilizador adiciona ou remove uma oferta de emprego aos seus favoritos, esta informação é enviada para o servidor. Se o utilizador aceder à sua conta noutro dispositivo este vai ter a lista atualizada dos seus favoritos no novo dispositivo.

5.2.7 Perfil

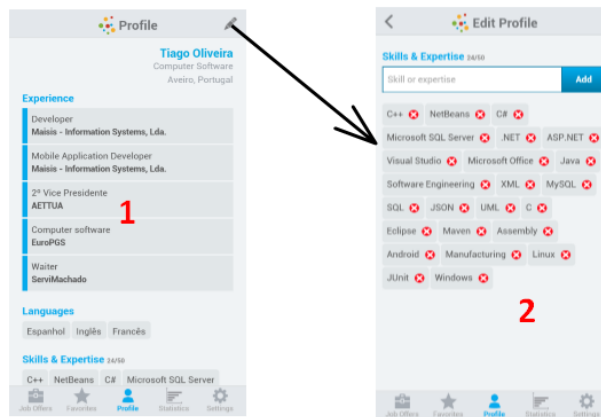


Figura 5.23: *Storyboard* do perfil de utilizador.

No ecrã 1 a informação a ser mostrada vai depender do tipo de registo que a pessoa fez na plataforma JOObian. Se for um registo feito pelo LinkedIn, a plataforma JOObian no processo de criação de um novo utilizador e fazendo uso das APIs do LinkedIn vai pedir informação do utilizador como a experiência profissional, os idiomas que domina e o seu perfil académico, de forma a enriquecer a informação mostrada no ecrã de perfil. Um utilizador registado sem recurso ao LinkedIn apenas verá a informação básica do perfil e as suas skills.

O utilizador também pode editar o seu perfil, para isso basta carregar no lápis que se encontra no canto superior direito do ecrã 1 e é encaminhado para o ecrã 2. Neste ecrã o utilizador pode adicionar novas skills para enriquecer o seu perfil ou então remover skills que não sejam essenciais para o seu perfil. O utilizador pode ir removendo skills ate atingir o mínimo de cinco skill, ao atingir este patamar é mostrado um alerta ao utilizador a indicar o mesmo.

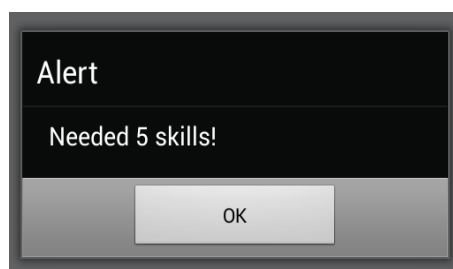


Figura 5.24: Exemplo do alerta ao atingir patamar mínimo de cinco skills.

O mesmo acontece quando o utilizador atinge o número máximo de skills (cinquenta). Se houver alteração do perfil o utilizador ao voltar à pilha de resultados é feito um pedido novo ao servidor por ofertas de emprego tendo em conta as novas alterações feitas.

5.2.8 Estatísticas

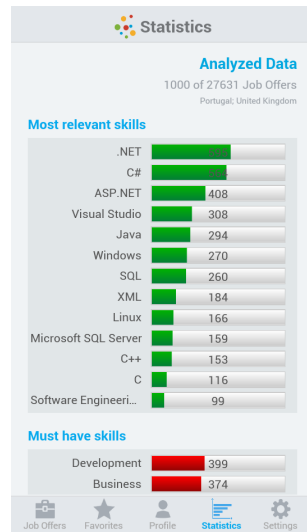


Figura 5.25: Ecrã de estatísticas do utilizador sobre as ofertas de emprego recomendadas.

Nesta ecrã são pedidos os dados estatísticos do perfil do utilizador sobre as ofertas de emprego. Ele pode ver das suas skills quais aquelas que mais estão presentes nas ofertas de emprego recomendadas para ele, bem como as que mais estão mais em falta ao seu perfil e que são muito pedidas pelos empregadores. Também pode ver uma lista das cidades que mais ofertas de emprego tem e quais as empresas que mais ofertas colocam no mercado.

Estes valores são obtidos a partir das ofertas de emprego recomendadas para aquele utilizador e tem em conta o “sorting” usado na pilha de ofertas de emprego. É mostrado ao utilizador o subset de ofertas avaliadas e quais os países de qual a estatística está a ser construída.

Esta página foi construída com o intuito de ser dinâmica, isto é, o número de resultados que aparece nos tópicos que a constitui é processado pelo servidor.



Figura 5.26: Ecrã de estatísticas do utilizador.

O calculo da percentagem da skill no subset de ofertas analisado foi o seguinte: (número de ofertas de emprego com a skill / número de resultados * 100).

O calculo da percentagem das cidades e empresas no subset teve que ser feito de outro método devido aos pequenos valores que estas tópicos representavam, a percentagem foi calculada da seguinte forma: (número de ofertas de emprego com a cidade ou empresa / Soma do número de ofertas de emprego com a cidade ou empresa * 100).

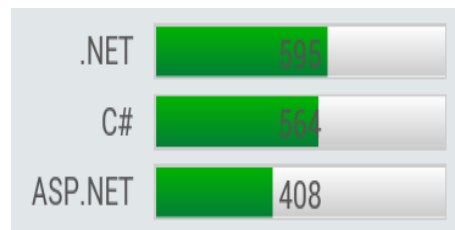


Figura 5.27: Valores representativos das estatísticas.

Para que o utilizador tive-se uma melhor representação dos valores decidiu-se que as barras representariam a percentagem da skill, cidade ou empresa e o número por cima desta barra representaria o número de ofertas de emprego com aquela skill, cidade ou empresa do subset de ofertas analisadas.

5.2.9 Definições

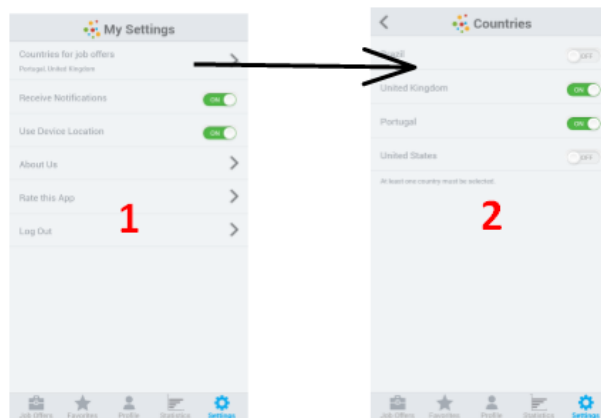


Figura 5.28: *Storyboard* das definições do utilizador na aplicação móvel JOObian.

O utilizador ao entrar no ecrã 1 tem uma lista com todas as definições da aplicação. O primeiro ponto da lista indica os países que o utilizador tem seleccionado no momento.

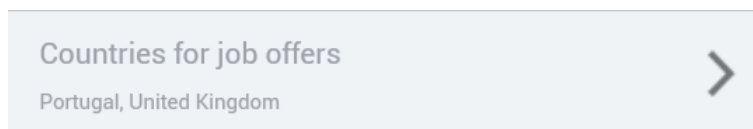


Figura 5.29: Lista de países seleccionados pelo utilizador.

A pilha de ofertas de emprego recomendadas que o utilizador recebe é sobre os países que este tem seleccionado. Ao clicar neste item da lista este é encaminhado para o ecrã 2. Neste ecrã o utilizador tem todos os países que a plataforma JOObian suporta no momento. Se o utilizador adicionar ou alterar os países para as ofertas de emprego esta informação é enviado ao servidor e são atualizadas as definições do utilizador na plataforma JOObian. Ao regressar ao ecrã da pilha de ofertas de emprego recomendadas é feito um pedido novo ao servidor por ofertas de emprego tendo em conta as alterações feitas pelo utilizador. O utilizador tem que ter no mínimo sempre um país seleccionado, não deixando remover um país se esse já foi o único seleccionado.

Os dois elementos seguintes da lista referem-se as notificações e ao uso da localização do dispositivo.

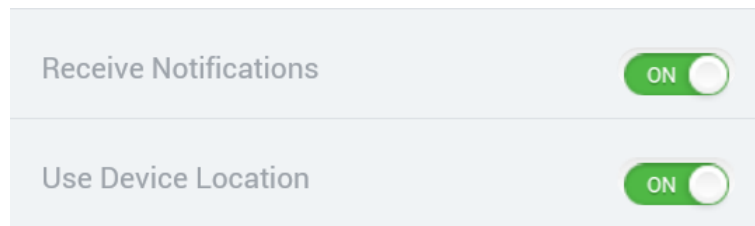


Figura 5.30: Definições: Notificações e uso da Localização.

O primeiro item da Figura 6.29, permite ativar ou desativar o envio de notificações da plataforma JOOBian. Estas notificações indicam o número de ofertas de emprego novas que o utilizador tem desde a sua ultima entrada na aplicação móvel JOOBian.

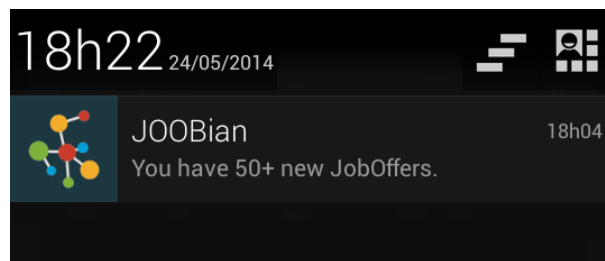


Figura 5.31: Notificação enviada pela plataforma JOOBian.

O segundo item da Figura 6.29 refere-se ao uso da localização do dispositivo. Se esta opção estiver ativada ao fazer-se “sorting” por proximidade é enviado ao servidor no pedido de ofertas de emprego a latitude e longitude do dispositivo móvel.

Caso o utilizador tenha feito login na aplicação sem fazer uso do LinkedIn o quarto item da lista é o “Change Password”. Ao carregar neste item o utilizador é encaminhado para o ecrã da Figura 6.31.

 A screenshot of a 'Change Password' screen. At the top, there is a back arrow and the title 'Change Password' next to a small logo. Below the title are three input fields: 'Old...', 'New. At least 6 characters', and 'Re-type new...'. At the bottom of the form is a blue button labeled 'Change password'.

Figura 5.32: Ecrã de alteração de palavra passe.

Neste ecrã o utilizador tem que inserir a palavra passe atual, uma nova palavra passe no mínimo com 6 carateres e confirmação da mesma. Todos estes campos são verificados e em

caso de erro o utilizador recebe um alerta a indica-lo do sucedido.

O próximo item da lista é o “About us”, o utilizador ao carregar neste item é encaminhado para o ecrã da figura 6.32.

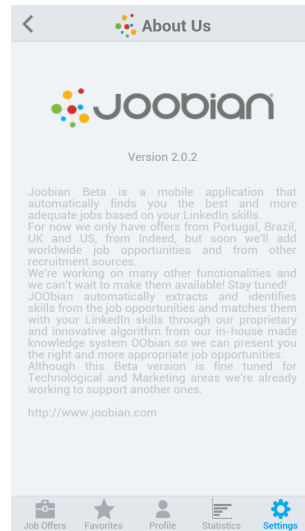


Figura 5.33: Ecrã “About us”.

Neste ecrã o utilizador pode ver qual é a versão e a descrição da aplicação.

O elemento que se segue na lista é o “Rate this App”, neste item é utilizado um plugin que nativamente abre a respetiva loja de aplicações do sistema onde é mostrada a aplicação móvel JOObian. Aqui o utilizador pode deixar um comentário sobre a aplicação móvel e atribuir uma classificação.

O último ponto da lista é o “Log Out”, o utilizador ao clicar neste item todos os dados em local storage são limpos e a aplicação móvel volta ao ecrã inicial.

Conclusão: Fazer todas a estilização e lógica da aplicação foi o que consumiu mais recursos e tempo. De salientar que toda informação mostrada no aplicação vem de serviços expostos pela plataforma JOObian.

5.3 Testes

Nesta capítulo vão ser analisados de uma forma geral, os vários testes realizados à aplicação móvel JOObian. Vão ser apresentados os testes de validação, que têm como objetivo assegurar o correto funcionamento da aplicação móvel no consumo dos dados externos. Serão também apresentados e analisados os testes de compatibilidade e usabilidade que tem como objetivo assegurar o correto funcionamento e visualização do conteúdo em diferentes dispositivos. No fim, é feita uma avaliação global dos resultados obtidos na realização dos testes.

5.3.1 Testes de validação e aceitação

Estes testes não são automatizados mas são bastantes importantes na medida que permitem verificar se os *uses cases* implementados funcionam como o esperado para o utilizador. Foram feitos testes exaustivos que tinham como objetivo assegurar o correto funcionamento da aplicação móvel quando esta manipula dados, quer estes sejam coerentes ou não. Dessa forma, após cada *release* a aplicação era instalada nos dispositivos móveis das equipas de desenvolvimento da empresa para efetuarem testes, antes de seguir para a equipa responsável pelos testes de qualidade do software.

5.3.2 Testes realizados

Os testes realizados à aplicação móvel JOObian incidiram sobre os requisitos funcionais identificados na secção 4.2. Estes requisitos controlam grande parte da aplicação e o seu correto funcionamento é fundamental para que esta funcione de acordo com as suas especificações.

Os testes aqui realizados incidiram sobre todos os componentes do HTML com o qual o utilizador pode interagir e sobre toda a lógica da aplicação. A título de exemplo, nestes testes foram passados *inputs* válidos e inválidos, e para que os testes tivessem sucesso era preciso que a lógica da aplicação soubesse gerir qualquer exceção interna para não comprometer o funcionamento global da mesma.

Temos por exemplo, no processo de “Sing up” ou “Sign in” onde é preciso validar os campos passados pelo utilizador. No caso do endereço de correio eletrónico é aplicada uma expressão regular para validar o mesmo; verificar se a palavra passe tem no mínimo seis caracteres; verificar se um utilizador já existir e este estiver no processo de “Sign up” encaminha-lo para o processo de “Sign in” e vice-versa; verificar se após um “Sign in” ou processo de registo os dados recebidos são os corretos para aquele utilizador.

Outro dos fatores mais críticos é a falta de conectividade, ou se por alguma razão os serviços demoram muito tempo a responder é necessário atribuir *timeouts* a estes pedidos.

A princípio vários destes processos não passaram nos testes, maioritariamente devido a cenários específicos que não tinham sido levados em consideração. Estes casos só foram detetados nos testes e à medida que iam surgindo eram analisados e as suas causas corrigidas, sendo que no fim todos os testes foram aprovados.

Apesar dos testes realizados terem sido exaustivos, não quer dizer que a aplicação não tenha qualquer falha, mas sim, que face aos testes realizados todas as falhas foram corrigidas.

No processo interno de qualidade, todos os produtos desenvolvidos na Maisis - Information Systems, Lda, têm um documento de plano de testes. Dado que este documento é na sua plenitude muito extenso não se considera relevante a sua inclusão total. Nos anexos deste documento seguem para título de exemplo alguns dos testes aplicados à aplicação móvel JO-Obian.

5.3.3 Testes de compatibilidade e usabilidade

Estes testes consistem em testar a aplicação nos mais variados *smartphones*, com o objetivo de assegurar o seu correto funcionamento e visualização. A utilização de vários dispositivos com sistemas operativos e versões diferentes deve-se ao facto de nem todos eles suportarem o mesmo conjunto de funcionalidades, portanto é preciso saber qual o impacto junto da aplicação móvel. Para além disto, também foi testada em ecrãs com diferentes resoluções, neste caso as mais utilizadas nos *smartphones*. Para ajudar nestes testes os emuladores que as plataformas disponibilizam tiveram um papel importante, pois permitem criar dispositivos virtuais com as características que forem necessárias, nomeadamente a dimensão e resolução de ecrã. Os testes foram realizados manualmente dando mais relevo à interface da aplicação. Estes testes são mais subjetivos ao contrário dos testes anteriormente descritos.

Foram utilizados os seguintes *smartphones* e *tablets* nos testes realizados:

- LG Optimus One P500
- Samsung Galaxy S
- ZTE A18
- Samsung Galaxy W
- BQ Aquaris 5
- LG Optimus 4X HD P880
- Motorola Moto G
- LG Nexus 5
- iPhone 4
- iPhone 5S
- Asus Nexus 7
- iPad Mini
- iPad - 1º geração

Os testes realizados nos *smartphones*, *tablets* e emuladores serviram numa primeira etapa para validar a correta visualização da aplicação nas diferentes resoluções de ecrã. A aplicação foi testada nas seguintes resoluções: 320x480, 480x800, 540x960, 600x905, 720x1080, 800x1280,

1080x1920. Nos iPhones e iPads não foram usadas resoluções mas sim o “aspect ratio” do dispositivo sendo estes os valores: 2/3, 40/71 e 3/4.

Todo o HTML foi desenvolvido sobre a resolução 720x1080 e apesar de todos os estilos usarem percentagens os resultados não foram os esperados pois os componentes do HTML ficavam fora do sitio em resoluções menores ou maiores. A solução passou por usar *media queries* no CSS para as resoluções e “aspect ratios” mencionados acima.

A segunda etapa consistiu em testar a aplicação de modo a aferir de que maneira é que aplicação era afetada pelas diferentes versões e sistemas operativos.

Para isso foram utilizados um conjunto de parâmetros para classificar a aplicação, estes foram os seguintes:

- Correto *layout* das páginas.
- Suporte a todos os estilos do CSS3 utilizados.
- Suporte às animações utilizadas nas transições das páginas.
- Suporte às *media queries* utilizadas no CSS3.
- Fluidez da aplicação.
- Correto funcionamento da lógica de negócio.

Dos parâmetros mencionados há três que se destacam na sua importância para a aplicação ser bem aceite junto dos utilizadores, sendo eles: o correto *layout* das páginas, a fluidez da interface e o correto funcionamento da lógica de negócio.

Nos estilos usados no CSS3 de salientar que nem todas as versões do Android aplicavam bem a propriedade *border-radius* porque em vez de aparecer cantos arredondados apareciam cantos retos. Também foi detetado no Android que ao aplicar uma cor com transparência esta passado alguns segundos perdia a transparência e ficava opaca. O impacto disto na aplicação é mínimo uma vez que se trata só de estilos do CSS.

No suporte às animações utilizadas na transição entre páginas não foi detetado qualquer problema, em todos os dispositivos funcionou corretamente, nos equipamentos com menos capacidade notou-se um pequena lentidão ao efetuar as transições.

Quanto ao *layout* das páginas já foi mencionado que foi necessário usar *media queries* no CSS para corrigir as posições dos componentes de HTML para as diversas resoluções e “aspect ratios”. O suporte das *media queries* no CSS3 tiveram um impacto positivo na representação das páginas. Para cada resolução ou “aspect ratios” foi criada uma *media query* para atribuir estilos com tamanhos fixos, para a correta posição dos componentes na página.

Quanto à fluidez da aplicação detetou-se que nos equipamentos mais fracos esta não era a melhor. Aqui descobriu-se que com uso intensivo da aplicação e nos equipamentos mais fracos, que estes reiniciavam por um excessivo uso de memória RAM, com isto descobriu-se que a aplicação tinha “memory leaks”.

Este problema deve-se ao uso do jQuery nas diretivas do AngularJS, porque o jQuery cria uma referência no DOM, sendo que o AngularJS só apaga a referência do DOM, deixando lá o objeto. Como o objeto não tem referência o “garbage-collected” não limpa o objeto da memória. Resolveu-se este problema fazendo o “destroy” do objeto quando este já não era necessário.

A lógica de negócio como seria de esperar apresentou um bom funcionamento em todos os dispositivos, pois está implementada no JavaScript e aqui não pode haver interpretações diferentes do código.

Capítulo 6

Conclusão

Na temática do desenvolvimento de aplicações móveis surgem diferentes conclusões, na melhor abordagem para a concretização destas, sendo algumas delas de carácter subjetivo e de interpretações diferentes.

Esta área nos últimos anos teve um forte crescimento devido à explosão dos *smartphones* e *tablets* existentes no mercado. Como os utilizadores cada vez mais querem mobilidade nos seus equipamentos e acesso à informação na hora, as empresas e os *developers* virão aqui um mercado com grande potencial de negócio que se tornou muito rentável com o passar do tempo. Apesar de todo o desenvolvimento o futuro desta área prevê ter um bom crescimento, não só porque a tecnologia está sempre a evoluir mas também pelo aparecimento de novos dispositivos no mercado. Temos por exemplo os relógios inteligentes ou o co-processador que a Apple e a Samsung implementaram nos seus dispositivos, abrindo assim as portas ao desenvolvimento de novas aplicações para dispositivos móveis.

O mercado atual tem um conjunto alargado de *frameworks* ou plataformas para o desenvolvimento de aplicações web ou híbridas, mas estas ainda não estão totalmente estáveis, estas ainda tem *bugs* e ainda não suportam todas as funcionalidades. As plataformas que estão assentes na tecnologia web, como HTML, JavaScript e CSS, tem que lidar com o fator dos navegadores ainda não suportarem todas as funcionalidades do HTML5. As empresas responsáveis por estes sistemas operativos têm feito um esforço enorme para dar suporte a todas as funcionalidades do HTML5 e a imagem seguinte mostra esse trabalho, onde de versão para versão o suporte a novas funcionalidades é maior.

O trabalho aqui descrito nesta tese mostra que existem muitos desafios quando se está a desenvolver uma aplicação híbrida, desde o tamanho e resoluções de ecrã dos diferentes dispositivos, na realização de testes bem como na fluidez da aplicação devido a esta correr na WebView do sistema operativo móvel. Estes são aspetos fulcrais para que a aplicação seja bem aceite junto dos utilizadores, sob pena de comprometer a aplicação desenvolvida.

	Android	BlackBerry	Chrome	Firefox	iOS	Opera	Windows Phone
Upcoming							8.1 372
Current	4.4 428	10.2 491	33 488	27 464	7.0 412	16 471	8 332
Older	4.0 272	7 275	32 484	26 462	6.0 387	15 441	7.5 128
	2.3 187		31 484	25 462	5.1 328	12.10 391	
			30 484	24 452	4.2 215	11.50 301	
			18 392	23 447	3.1 131		
				22 457			

Figura 6.1: Número de funcionalidades suportadas pelos navegadores dos dispositivos móveis, das 555 funcionalidades do HTML5 em cada versão do sistema operativo ou navegador móvel.[Nie]

6.1 Projeto Desenvolvido

A aplicação móvel desenvolvida deu suporte a todos os requisitos levantados na secção 4.2, ficando por implementar os de menor prioridade, pois esses ainda não se encontravam desenvolvidos do lado do servidor. A aplicação tinha como objetivo encontrar as ofertas de emprego que melhor se enquadravam com o perfil do utilizador fazendo uso das suas skills e área de estudo. Esta parte foi concluída com sucesso, ficando a gestão de carreira para um trabalho futuro.

Com o desenrolar do projeto e da perspetiva de quem já desenvolveu algumas aplicações nativas tanto para Android como iOS, foi notório o esforço acrescido para o desenvolvimento da aplicação em comparação com o desenvolvimento nativo. Este esforço incidiu sobre os *layouts* em HTML, no uso da *framework* AngularJS e nos testes da aplicação, pois não era possível fazer Debug ao código JavaScript em “RunTime”.

Apesar destes entraves a escolha por este método para o desenvolvimento da aplicação móvel JOObian, revelou-se perfeitamente adequado aos objetivos e requisitos definidos. Maior esforço seria necessário se aplicação fosse desenvolvida nativamente para cada sistema operativo móvel.

6.2 Apreciação pessoal

Ter realizado a tese em ambiente empresarial foi uma mais valia para mim, pois foi uma oportunidade de mostrar e testar os meus conhecimentos adquiridos ao longo da vida académica, ter contacto com o mundo de trabalho na área do desenvolvimento de software e ganhar novas skills para enfrentar novas oportunidades de emprego.

Aprendi com os erros devido a falta de experiência e conhecimento, pelo que se me fosse dada uma nova oportunidade para recomeçar o projeto seria capaz de realizar um trabalho com uma qualidade superior.

O objetivo pessoal de ter algo desenvolvido por mim disponível nas lojas de aplicações

móveis foi concretizado e foi uma mais valia ter contribuído positivamente para que houvesse pessoas a encontrar emprego através da aplicação e poder obter *feedback* dessa utilização.

No que toca ao tema abordado nesta dissertação a minha opinião cai sempre em desenvolver nativamente, apesar do esforço que isso acarreta para uma empresa ou *developers* em termos de tempo e dinheiro. Deparei-me muitas vezes com código desenvolvido que funcionava corretamente em iOS e em Android corria tudo mal. Outro dos problemas que me fez ficar apreensivo com a utilização destas plataformas teve a ver com o facto das resoluções dos dispositivos, pois tive que replicar os estilos no CSS para cada resolução, compilar e correr a aplicação centenas de vezes até acertar com as posições para cada dispositivo. O facto que não poder fazer *debug* ao código JavaScript foi um ponto muito negativo destas plataformas.

Contudo não vi só aspetos negativos nesta abordagem, estas plataformas fornecem de uma forma muito simples o acesso aos recursos do dispositivo e acesso a uma comunidade onde já existe muitos plugins desenvolvidos para utilizar na aplicação. O facto de desenvolver só uma vez e gerar uma aplicação para as diferentes plataformas fez com que poupasse bastante tempo no desenvolvimento da mesma.

6.3 Trabalho Futuro

É também importante apontar o trabalho que há de ser feito posteriormente ao termino do âmbito da tese, pois é preciso cativar novos utilizadores e manter os antigos lançando novas funcionalidades na aplicação móvel. Como já foi referido na secção 8.1 a gestão de carreira iria ficar para uma implementação futura pois essa parte ainda não estava implementada do lado do servidor.

Outros prontos a implementar são:

- Partilhar a oferta de emprego por SMS, correio eletrónico, Facebook, Twitter, Google+, entre outros.
- Possibilidade de exportar o CV do utilizador.
- Possibilidade de editar o nome, cidade, país e indústria do utilizador.
- Possibilidade de editar os pesos das skills, para dar mais relevo a determinadas skills que outras.
- Ver a oferta salarial da oferta de emprego.
- Ver o custo de vida da cidade da oferta de emprego.
- Possibilidade de aplicar filtros avançados sobre as ofertas de emprego.
- Ter um mapa com as localizações das ofertas de emprego.
- Criar uma aplicação web para utilizadores que não tenham *smartphones* ou *tablets*

Outro objetivo futuro complementar aos pontos anteriores poderá passar por disponibilizar a aplicação em outros sistemas operativos com alguma expressão de mercado, nomeadamente Windows Phone e ou Blackberry, para os quais também existe suporte da plataforma PhoneGap.

Bibliografia

- [Ma12] Marco Casario, Nathalie Wormser, Dan Saltzman, Anselm Bradford, Jonathan Reid, Francesco Improta, Aaron Congleton. *CSS3 Solutions: Essential Techniques for CSS3 Developers*. Apress, 2012.
- [Adr13] Adrian Kosmaczewski. *Sencha Touch 2 Up and Running*. O'Reilly Media, Inc., 2013.
- [And12] Andrew Trice. Phonegap explained visually. [consult. 20 jan. 2014]. <http://phonegap.com/2012/05/02/phonegap-explained-visually/>, May 2012.
- [And14] Android. Dashboard - platform versions. [consult. 17 jun. 2014]. https://developer.android.com/about/dashboards/index.html?utm_source=ausdroid.net, 4 June 2014.
- [App14] Apple Developer. Apple developer support. [consult. 17 jun. 2014]. <https://developer.apple.com/support/appstore/>, 15 June 2014.
- [Bra11] Brad Broulik. *Pro jQuery Mobile*. Apress, 2011.
- [Bra13] Brad Green, Shyam Seshadri. *AngularJS*. O'Reilly Media, Inc., 2013.
- [Bru13] Bruno Cavaleiro. *Easy Recruitment Plataforma de Recrutamento*. Faculdade de Ciências e Tecnologia da Universidade de Coimbra, 2013.
- [Cli14] Cliff Kimani. 7 cross-platform mobile development tools. [consult. 21 nov. 2013]. <https://www.udemy.com/blog/cross-platform-mobile-development/>, February 2014.
- [Dav10] David, Matthew. *HTML5 Designing Rich Internet Applications*. CompX Digital, 2010.
- [Dav11] David Sawyer McFarland. *JavaScript & jQuery: The Missing Manual, Second Edition*. O'Reilly Media, 2011.
- [Dav12] David Power. *Beginning CSS3: Mastering the Language of Web Design*. Apress, 2012.
- [Eag08] Eaglesham, J. Scrum. [consult. 4 abril. 2014]. <http://epf.eclipse.org/wikis/scrum/>, 2008.
- [Ecm13] Ecma International. *Standard ECMA-404 - The JSON Data Interchange Format, 1st Edition*. Ecma International, 2013.

- [Eri95] Erich Gamma, Richard Helm, Ralph E. Johnson, John Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, Reading, MA, 1995.
- [IDC13] IDC. Android and ios combine for 92.3% of all smartphone operating system shipments in the first quarter while windows phone leapfrogs blackberry, according to idc. [consult. 13 mar. 2014]. <http://www.idc.com/getdoc.jsp?containerId=prUS24108913>, May 2013.
- [Joh13] John M. Wargo. *Apache Cordova 3 Programming*. Addison-Wesley, 2013.
- [M13] Mrcio Souza. Angularjs o que é? [consult. 3 fev. 2014]. <http://www.kazale.com/angularjs-o-que-e/#more-823>, Dezembro 2013.
- [Mar] Marc Plotz. Introduction to jquery mobile. [consult. 20 nov. 2013]. http://www.htmlgoodies.com/html5/tutorials/introduction-to-jquery-mobile.html#fbid=_WdjuN-FWJs.
- [Nie] Niels Leenheer. how well does your browser support html5? [consult. 25 mai. 2014]. <http://html5test.com/results/mobile.html>.
- [Phoa] PhoneGap. Phonegap documentation. [consult. 15 jan. 2014]. http://docs.phonegap.com/en/edge/guide_cli_index.md.html#The%20Command-Line%20Interface.
- [Phob] PhoneGap. Supported features. [consult. 15 jan. 2014]. <http://phonegap.com/about/feature/>.
- [Pri] Priya Viswanathan. Top 5 tools for multi-platform mobile app development. [consult. 21 nov. 2013]. <http://mobiledevices.about.com/od/mobileappbasics/tp/Top-5-Tools-Multi-Platform-Mobile-App-Development.htm>.
- [Ral13] Raluca Budiu. Mobile: Native apps, web apps, and hybrid apps. [consult. 3 dez. 2013]. <http://www.nngroup.com/articles/mobile-native-apps/>, September 2013.
- [Sta] StackExchange. About cross-platform. [consult. 2 jun. 2014]. <http://programmers.stackexchange.com/tags/cross-platform/info>.
- [The] The jQuery Foundation. What is jquery? [consult. 13 jun. 2014]. <http://jquery.com/>.
- [The12] Thejaswi Puthraya. Dissecting phonegap's architecture. [consult. 15 fev. 2014]. <https://agiliq.com/blog/2012/09/dissecting-phonegaps-architecture/>, September 2012.
- [Unk12] UnknownCom Inc. *Producing iOS 6 Apps: The Ultimate Roadmap for Both Non-Programmers and Existing Developers*. UnknownCom Inc., 2012.
- [Xam14] Xamarim. Introduction to mobile development - understanding xamarin mobile application projects. [consult. 25 nov. 2013]. http://docs.xamarin.com/guides/cross-platform/getting_started/introduction_to_mobile_development/, 2014.

Apêndice A

Documento de Requisitos

JOOBIAN

Aplicação Móvel

Requisitos

11/06/2014

5.0

Tiago Oliveira
Miguel Grade

Histórico de versões do documento de Requisitos

Versão	Datas	Responsabilidades	Justificação
1.0	02/10/2013	Edição: Miguel Grade	Versão Inicial
	02/10/2013	Revisão: Filipe Augusto	
	02/10/2013	Aprovação: Filipe Augusto	
2.0	10/12/2013	Edição: Tiago Oliveira	Favoritos Ordenação de Ofertas Pesquisa de Ofertas
	10/12/2013	Revisão: Miguel Grade	
	10/12/2013	Aprovação: Filipe Augusto	
3.0	13/02/2014	Edição: Tiago Oliveira	Partilha de Ofertas Notificações
	13/02/2014	Revisão: Miguel Grade	
	13/02/2014	Aprovação: Filipe Augusto	
4.0	01/04/2014	Edição: Tiago Oliveira	Registo de Utilizador Estatísticas
	01/04/2014	Revisão: Miguel Grade	
	01/04/2014	Aprovação: Filipe Augusto	
5.0	11/06/2014	Edição: Tiago Oliveira	Múltiplos Providers de Ofertas Edição de Perfil
	11/06/2014	Revisão: Miguel Grade	
	11/06/2014	Aprovação: Filipe Augusto	

Índice

1.	Introdução.....	5
1.1.	Documentos Relacionados	5
1.2.	Estrutura do Documento	5
2.	Descrição do Sistema.....	6
2.1.	Motivação e Objetivos	6
2.2.	Público-alvo.....	7
2.3.	Plataforma JOOBIAN	8
3.	Atores e Personas	10
3.1.	Utilizador.....	10
3.1.1.	Pedro Antunes	11
3.1.1.	Ana Silva	11
3.1.2.	Rui Neves.....	12
3.1.3.	João Martins.....	13
4.	Requisitos Funcionais.....	14
4.1.	Ecrã Inicial	14
4.2.	Registar na Aplicação – Passo 1	15
4.3.	Mensagens de Erro – Registo – Passo 1	16
4.4.	Registar na Aplicação – Passo 2	17
4.5.	Mensagens de Erro – Registo – Passo 2	18
4.6.	Registar na Aplicação – Passo 3	19
4.7.	Mensagens de Erro – Registo – Passo 3	20
4.8.	Registar na Aplicação – Passo 4	21
4.9.	Registar na Aplicação – Passo 4 – Escolha de Países	22
4.10.	Registar na Aplicação via LinkedIn.....	23
4.11.	Autenticação na aplicação.....	24
4.12.	Mensagens de Erro – Autenticação na aplicação	25
4.13.	Autenticar na aplicação através do LinkedIn	25
4.14.	Menu da aplicação	26
4.15.	Listagem de ofertas de emprego	27
4.16.	Ordenação da Listagem da Oferta de Emprego	28
4.17.	Pesquisa de Ofertas de Emprego	29
4.18.	Resultados de Pesquisa	30
4.19.	Detalhes da Oferta de Emprego.....	31
4.20.	Adicionar/Remover Oferta de Emprego aos Favoritos	32
4.21.	Matched Skills e Missing Skills	32
4.22.	Idiomas da Oferta	33
4.23.	Descrição da Oferta.....	33
4.24.	Ver oferta de emprego na fonte da oferta	33

4.25.	Partilhar Oferta de Emprego.....	34
4.26.	Favoritos.....	35
4.27.	Visualização de Perfil do Utilizador	36
4.28.	Edição de Perfil do Utilizador.....	37
4.29.	Estatística.....	39
4.30.	Settings	40
4.31.	Escolha de Países.....	41
4.32.	Receber notificações	42
4.33.	Utilizar a localização atual	42
4.34.	Mudar palavra-passe	43
4.35.	Sobre a aplicação.....	44
4.36.	Avaliar a aplicação	45
4.37.	Sair da aplicação.....	45
4.38.	Termos de Serviço e Política de Privacidade.....	46
4.39.	Esqueci Password.....	47
4.40.	Hardware do dispositivo	48
4.41.	Requisitos de performance.....	48
4.42.	Recuperação do sistema em caso de falha	48
4.43.	Ver oferta salarial.....	49
4.44.	Ver Index salário/custo de vida	49
4.45.	Importar CV	49
5.	Listagem de Requisitos	50
6.	Anexos.....	53
6.1.	Termos de Serviço e Política de Privacidade.....	53

1. Introdução

Os requisitos definidos neste documento são referentes à implementação de uma plataforma de emprego a ser desenvolvida denominada Joobian. Inicialmente é feita uma descrição do sistema e das componentes envolvidas e em seguida são apresentadas as motivações e objetivos da plataforma que levam a que haja interesse na sua criação.

Para ajudar na definição dos requisitos a implementar foram criados vários atores e personas que definem os vários intervenientes no sistema. De seguida são apresentados os requisitos funcionais do sistema. Estes encontram-se descritos na forma de casos de uso.

1.1. Documentos Relacionados

Não existem documentos relacionados.

1.2. Estrutura do Documento

O presente documento discrimina o conjunto de requisitos identificados para uma primeira versão funcional da aplicação móvel.

Inicialmente é feita uma breve descrição da aplicação, seguindo-se a identificação e especificação discriminada dos requisitos a implementar.

2. Descrição do Sistema

A aplicação móvel Joobian pretende efetuar o cruzamento das ofertas de emprego existentes no mercado com o perfil de cada utilizador, possibilitando dessa forma que este último obtenha as ofertas de trabalho que melhor se adequam ao seu perfil.

O cruzamento efetuado tem como parâmetros os diversos pontos do perfil do utilizador e as diferentes características da oferta. De uma forma inicial tudo pode ser efetuado de forma semiautomática ao seja com a interligação com o LinkedIn onde o utilizador já definido o seu perfil e com a análise ao texto da oferta esse cruzamento é possível. No entanto para que a garantia de bons resultados seja cada vez maior o utilizador poderá alterar parâmetros no seu perfil, tentando que as ofertas sejam mais adequadas ao mesmo.

2.1. Motivação e Objetivos

A plataforma de emprego Joobian será construída com o intuito de ter o maior número de dados tanto de perfis de candidatos a emprego como de ofertas de emprego pois, quantos mais dados houver na plataforma, mais valor será possível gerar através desta. Tendo este como o principal objetivo, é necessário que a aplicação/cliente móvel aqui detalhado maximize esta estratégia, para isso é vital cumprir com os seguintes pontos-chave:

- Facilitar e incentivar que mais pessoas se juntem à plataforma e registem os seus dados – Assim será possível fazer com que mais pessoas (e empresas) adiram à plataforma e registem nela os seus dados;
- Possibilidade de integração com sistemas já existentes – Havendo integração com outros sistemas, é possível atrair também empresas que já utilizem outros sistemas para este efeito.
- Utilização de fontes externas que permitam obter este tipo de dados – Isto permite a existência prévia de alguns dados no sistema quando for inicializado, não havendo dependência de adição de dados diretamente por utilizadores, e uma maior variedade destes devido à obtenção de várias fontes (e.g. dados do Curriculum extraídos através do LinkedIn ou CV Europass)

2.2. Público-alvo

Como já referido anteriormente, os principais utilizadores desta plataforma serão:

- Candidatos: Pessoas que estejam interessadas em consultar informação sobre ofertas de emprego, sendo estes a principal fonte de perfis de candidatos.
- Empresas: Empresas que tencionam encontrar candidatos para as ofertas de emprego que possuem, e das quais serão a principal fonte.

Note-se que para este componente que estamos a levantar requisitos (Joobian cliente móvel) iremos nos focar nos Candidatos.

Tendo em conta que cada um destes adiciona ao sistema os dados que interessam ao outro, há aqui a criação de um efeito de bola de neve que vai assegurar o incremento do valor da plataforma ao longo do tempo (**Erro! A origem da referência não foi encontrada.**).



Figura 1 - Representação do efeito de bola de neve gerado

Numa segunda fase, tendo dados suficientes presentes no sistema, será possível fazer uma análise destes que permita oferecer aos seus utilizadores um serviço mais direcionado às suas necessidades através de recomendações e métricas obtidas da análise. Isto irá permitir uma correspondência entre candidatos e ofertas de emprego.

2.3. Plataforma JOOBIAN

A plataforma Joobian irá fornecer serviços de apoio ao cliente móvel que iremos desenvolver e detalhar ao longo deste documento, sendo que para entender melhor que é a plataforma Joobian iremos dar uma visão geral desta plataforma de recrutamento que terá então (conforme descrito no capítulo anterior) que ser capaz de interagir e obter dados de várias fontes externas.



Figura 2 - Vista global da plataforma e conexões com fontes externas.

Como podemos identificar na figura acima, há quatro fontes que poderão ser utilizadas para introdução direta ou indireta de dados no sistema.

- Portais de emprego: Estes serão utilizados para obter as últimas ofertas publicadas e possíveis de obtenção por feeds.
- Currículos Europass: Para facilitar o registo dos dados curriculares do utilizador, este poderá utilizar um currículo, formato Europass, para preencher campos do formulário de registo.
- LinkedIn: Em vez do currículo, descrito em cima, um utilizador pode também utilizar a informação do seu perfil do LinkedIn para o mesmo efeito, pois este funciona como um currículo *online*.
- Outras Plataformas: Pode também haver integração com outras plataformas de modo a permitir que empresas que já utilizem outros sistemas possam utilizar serviços disponibilizados pela nossa plataforma.

A vantagem de ter duas fontes para obter informação para preenchimento de dados do utilizador é a existência de possíveis utilizadores que podem ter apenas uma das duas, assim a probabilidade de um destes conseguir importar os dados de outro sítio onde já estejam preenchidos é maior.

Feeds RSS:

Os portais de emprego identificados que poderão ser utilizados como fontes de feeds foram o Sapo emprego, o Expresso emprego e o ITJobs. Como estes feeds devolvem um ficheiro XML e para cada uma das fontes o esquema é diferente, é necessário considerar a utilização de uma configuração para cada fonte assim como a possibilidade de alteração do

esquema por questões de manutenção e adição de novas fontes. Para isto deverá ser possível configurar genericamente qualquer fonte de feeds que se pretenda utilizar.

Como vão sendo lançadas novas ofertas de emprego nos feeds, esta extração deve ser feita periodicamente de maneira a que estas ofertas vão sendo adicionadas à plataforma.

Currículos Europass:

Estes vão ser utilizados na altura do registo ou edição de um perfil de utilizador para ajuda no preenchimento do formulário do perfil. É necessário que o utilizador carregue o ficheiro para a plataforma, que terá que ser em formato PDF. Depois de ser carregado o sistema terá que ler o mesmo e preencher os campos do formulário com a informação que for possível extrair.

É necessário considerar as várias versões existentes do formato Europass, pois cada versão tem um esquema e modelo de dados diferente e tem que ser tratada de maneira diferente, assim como as várias línguas. Também aqui terá que haver uma configuração que permita definir isto.

LinkedIn:

Os dados de perfil do LinkedIn, tal como os currículos, podem ser utilizados no registo e edição de um perfil de candidato. Este acesso é feito através da API do LinkedIn. Para isto é necessário ter uma aplicação registada no site de programador do LinkedIn¹, pois é preciso as credenciais obtidas neste registo para aceder autenticar o acesso à API. É ainda necessário que o utilizador que queira utilizar este método de extração conceda autorização para tal, o que resulta na obtenção de um *access token* que é utilizado para aceder aos dados do perfil do mesmo.

Como após ter o *access token* do utilizador é possível aceder à informação do seu perfil posteriormente, a não ser que expire a sua validade ou o utilizador termine a autorização dada, é pretendido fazer uma atualização periódica destes dados de modo a que o utilizador, ao atualizar o seu perfil no LinkedIn, este seja automaticamente atualizado na plataforma.

Outras Plataformas:

Para questões de integração do sistema com outras plataformas que os clientes utilizem, será disponibilizado uma API sob a forma de um serviço web RESTful. Este serviço devolverá respostas em JSON e permitirá que os clientes da plataforma tenham ao seu dispor as mesmas funcionalidades que teriam acedendo à aplicação web, a não ser que tal não seja possível por questões de visualização.

¹ <https://developer.linkedin.com/>

Além destas funcionalidades descritas o sistema deve disponibilizar consulta de ofertas de emprego e perfis de candidatos através de pesquisa, por palavras-chave e filtros, e recomendações geradas através de análise de dados.

Este deve também ter em conta a possível adição de mais módulos no futuro que possibilitem integração com outros tipos de sistemas, por isso deverá seguir uma arquitetura modular.

3. Atores e Personas

Neste capítulo são apresentados alguns dos possíveis atores/personas que poderão usar o Joobian, este tópico torna-se interessante para poder contextualizar a utilização da aplicação e para poder definir o mais corretamente possível os requisitos.

Para cada persona é apresentado:

- Descrição – Descrição sumária da pessoa (idade, nível de estudos, localidade, profissão/área profissional).
- Objetivos pessoais – Os objetivos pessoais da persona que a podem levar a utilizar a plataforma.
- Objetivos para o sistema – objetivos, tendo em conta os seus objetivos pessoais, que esta terá ao utilizar a plataforma. Estes são objetivos mais palpáveis que podem ser traduzidos diretamente em casos de uso. Estes são divididos em primários e secundários, sendo os primários correspondentes a objetivos que podem ser conseguidos por tarefas que o utilizador desempenha por ele

3.1. Utilizador

Este grupo pode representar pelo menos três subgrupos de pessoas:

- **Candidato:** Utilizador, desempregado ou não, que se encontre no momento à procura de ofertas de emprego
- **Profissional:** Utilizador com uma carreira estável que queira estar mais informado sobre o mercado da sua área e obter informação que lhe permita continuar a construir a sua carreira.
- **Mundial:** Utilizador que pretende obter informações de oportunidades em vários países, que não se prende e que deseja obter as melhores oportunidades sem olhar a distâncias.

3.1.1. Pedro Antunes

Descrição

O Pedro é um jovem de 22 anos, recém-licenciado em Engenharia do Ambiente pela Faculdade de Ciências e Tecnologias da Universidade de Coimbra.

Objetivos Pessoais

No fim do curso decidiu saber a oferta atual de trabalho na sua área para assim tomar a decisão de continuar os seus estudos e saber melhor que tipo de formação é mais pedida, ou, no caso de encontrar ofertas de trabalho interessantes, tentar a sua sorte numa destas e fazer a sua entrada no mercado de trabalho.

Objetivos para o Sistema

Para isto, ao utilizar a plataforma poderá pesquisar por ofertas de emprego na sua área e consultá-las. Pode também ter acesso a uma lista de ofertas recomendadas tendo em conta a informação do seu perfil e formação recomendada consoante o emprego em que esteja interessado.

- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação de oferta de emprego;
- Ver sugestões de emprego;
- Ver formação recomendada.

3.1.1. Ana Silva

Descrição

A Ana tem 25 anos, é natural de Viseu, tirou o Mestrado em Engenharia Informática na Faculdade de Ciências e Tecnologia da Universidade de Lisboa e encontra-se atualmente a trabalhar numa empresa de desenvolvimento de software na mesma cidade.

Objetivos Pessoais

O seu posto de trabalho permite-lhe ter uma vida estável, mas, devido à distância que a separa da família e dos amigos, num futuro próximo a Ana gostava de encontrar trabalho mais próximo da sua terra natal. No entanto gostaria de, pelo menos, manter um salário que lhe permita continuar a ter o estilo de vida atual no seu novo posto.

Objetivos para o Sistema

Ao utilizar o sistema a Ana pode procurar ofertas de emprego na sua área e filtrá-las por localização. Pode ainda consultar uma lista de empregos recomendados consoante a sua formação e localização. Para ter uma ideia do salário que poderá ficar a ganhar se mudar para estar mais perto de casa pode também consultar métricas que lhe mostrem uma média de quanto ganha alguém nessas condições.

- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação de oferta de emprego;
- Ver sugestões de emprego;
- Ver métricas de ajuda à gestão de carreira.

3.1.2. Rui Neves

Descrição

O Rui tem 29 anos e trabalha há 6 anos como web developer na mesma empresa.

Objetivos Pessoais

Ele gosta do seu posto de trabalho na empresa mas ao mesmo tempo sente que não está a progredir na sua carreira. Decide então procurar habilitações (skills) que o ajudem a alcançar isto e chegar a gestor de projetos da sua área.

Objetivos para o Sistema

Ao utilizar a plataforma este utilizador pode procurar por ofertas de emprego semelhantes ao cargo que ele quer atingir e ter acesso a sugestões de que tipo de formação e habilitações são necessários para atingir um cargo deste tipo.

- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ver informação de oferta de emprego;
- Ver formação recomendada.

3.1.3. João Martins

Descrição

O João tem 32 anos e trabalha há 8 anos como Engenheiro Eletrotécnico já passou por várias empresas em diversas cidades e países da Europa.

Objetivos Pessoais

Ele gosta dos desafios, tem uma necessidade de crescer profissional e salarialmente, gosta de conhecer novas cidades e culturas. No entanto decidiu que chegou uma nova etapa e decide então procurar melhorar não só a sua condição salarial mas também a qualidade de vida/poder de compra e necessita de saber qual será não só o próximo emprego mas também a sua próxima cidade/país.

Objetivos para o Sistema

Ao utilizar a plataforma este utilizador pode obter recomendações de ofertas de emprego direcionadas e alinhadas com o seu perfil, sendo que as mesmas deverão ter em conta as restrições por ele impostas, nomeadamente a maximização do binário salário/custo de vida.

- Pesquisar ofertas de emprego;
- Filtrar pesquisa de ofertas de emprego;
- Ordenar as ofertas de emprego por (Salário e/ou Index Custo de Vida)
- Ver informação de oferta de emprego;
- Ver índice de qualidade de vida

4. Requisitos Funcionais

4.1. Ecrã Inicial

ID do requisito: R.JOO.MOBAPP.01.v1

Dependências:

Descrição:

Quando o utilizador acede à aplicação pela primeira vez ou após efetuar a ação de *logout* a aplicação deve averiguar a conectividade do utilizador à internet. Caso não exista conectividade a aplicação deve mostrar uma mensagem com o texto "No internet connection!", caso contrário a aplicação deve apresentar 2 opções na parte inferior do ecrã. No lado esquerdo deve ser dada a possibilidade do utilizador se registar para ter acesso à aplicação enquanto do lado direito é dada a possibilidade de o utilizador se autenticar na aplicação. Estas opções devem ser apresentadas sob a forma de botões de acordo com o apresentado na Figura 3.

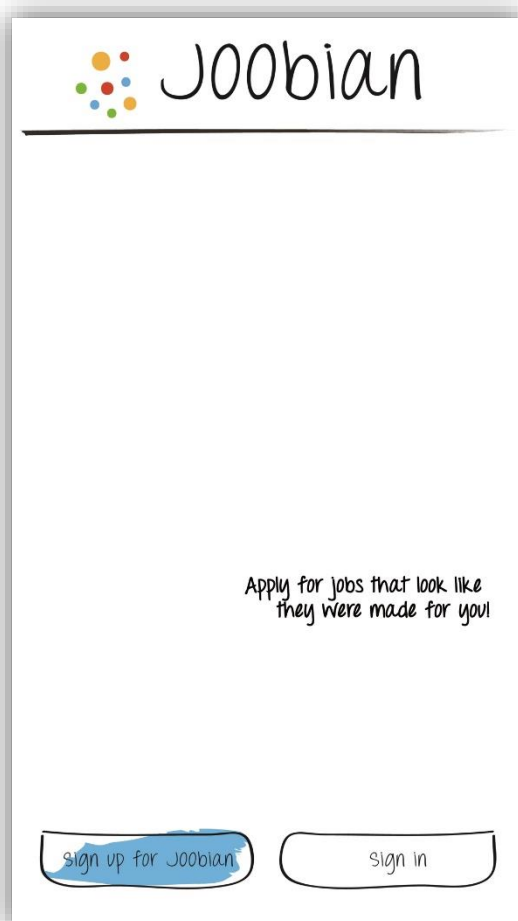


Figura 3 - Ecrã Inicial

4.2. Registrar na Aplicação – Passo 1

ID do requisito: R.JOO.MOBAPP.02.v1

Dependências: R.JOO.MOBAPP.01.v1

Descrição:

Ao clicar na opção de “Sign Up for Joobian” no ecrã inicial o utilizador deve ser remetido para o ecrã com o 1º passo do registo de acordo com o apresentado na Figura 4.

Neste ecrã é dada a possibilidade de o utilizador fazer um registo diretamente no Joobian ou de se registar via LinkedIn, clicando no botão que aparece no topo do ecrã.

Ao optar por se registar diretamente no Joobian o utilizador deverá preencher os campos de correio eletrónico, password e confirmação de password.

Na parte inferior do formulário devem ser mostrados 2 botões. Do lado esquerdo um botão de “Cancel” que permite ao utilizador cancelar o processo de registo remetendo o mesmo para o ecrã inicial. Do lado direito um botão de “Done” que permite ao utilizador passar ao próximo passo de registo.

Na parte inferior do ecrã deve ser mostrado o texto “Terms of Service & Privacy Policy” que deverá funcionar sob a forma de ligação para o ecrã com os Termos de Serviço e Política de Privacidade.

The image is a hand-drawn sketch of a mobile application registration screen. At the top, there is a logo consisting of several colored dots followed by the word 'Joobian'. Below the logo is a button with the LinkedIn logo and the text 'Log in with LinkedIn'. Underneath this button, the text 'or sign up with email address' is written. A large rounded rectangle contains three input fields: 'Email' with the placeholder 'Your email address', 'Password' with the placeholder 'At least 6 characters', and 'Confirm' with the placeholder 'Confirm Password'. Below these fields are two buttons: 'Cancel' on the left and 'Done' on the right. At the bottom of the screen is a link labeled 'Terms of Service & Privacy Policy'.

Figura 4 - Registo - Passo 1

4.3. Mensagens de Erro – Registo – Passo 1

ID do requisito: R.JOO.MOBAPP.03.v1

Dependências: R.JOO.MOBAPP.02.v1

Descrição:

A aplicação deverá verificar a conformidade do correio eletrónico inserido bem como a extensão mínima de 6 carateres para a password e a igualdade da password inserida no campo password e confirmação.

Caso o utilizador não preencha o formulário de acordo com o estipulado deve aparecer uma mensagem em formato de alerta com o texto relativo ao problema.

Caso o utilizador não tenha inserido nenhum texto no campo de email deve aparecer a mensagem com o texto "Email address needed!".

Caso o utilizador tenha inserido um email não conforme deve aparecer uma mensagem com o texto "You have entered na invalid email address!".

Caso a password inserida não tenha os 6 carateres necessários deve aparecer a mensagem "Your password has not 6 characters!".

Caso a password e a confirmação de password não sejam exatamente iguais deve ser apresentada a mensagem com o texto "The passwords are not the same!".

Caso o email indicado já esteja registado no joobian deverá ser apresentada a mensagem com o texto "User already exists. Please sign in..." e o utilizador deve ser redirecionada para o ecrã de autenticação.

Para fechar cada uma destas mensagens bastará ao utilizador clicar no botão de "OK".

4.4. Registrar na Aplicação – Passo 2

ID do requisito: R.JOO.MOBAPP.04.v1

Dependências: R.JOO.MOBAPP.02.v1

Descrição:

Ao clicar no botão “Done” do primeiro passo do registo e caso o formulário seja corretamente validado o utilizador deve ser remetido para o segundo passo do registo que deve ser apresentado de acordo com a Figura 5.

Neste ecrã o utilizador deve preencher os campos “First Name”, “Last Name” e “City” em campos de texto livre com o máximo de N carateres. O utilizador deverá de igual forma escolher o país em que se encontra e a indústria na qual desenvolve a sua atividade profissional.

Estas duas listagens devem incluir um vasto conjunto de possibilidades tentando abranger todo o globo e todas as áreas de atuação.

Na parte inferior da página devem aparecer 2 botões. Do lado esquerdo um botão com o Texto “Cancel” permitirá ao utilizador cancelar o processo de registo. Do lado direito um botão com o texto “Next 1/3” permite ao utilizador saltar para o próximo passo do registo.

The image is a hand-drawn sketch of a mobile application screen for the 'Joobian' app. At the top, there is a logo consisting of five colored dots (yellow, red, blue, green, and purple) arranged in a circle, followed by the word 'Joobian' in a handwritten-style font. Below the logo, there are four input fields arranged vertically. The first two are side-by-side: 'First Name' and 'Last Name'. The third is 'City'. The fourth is 'Choose Country' with a small downward arrow indicating a dropdown menu. Below that is 'Choose Industry' with a similar downward arrow. At the bottom of the screen, there are two buttons: 'Cancel' on the left and 'Next 1/3' on the right.

Figura 5 - Registo - Passo 2

4.5. Mensagens de Erro – Registo – Passo 2

ID do requisito: R.JOO.MOBAPP.05.v1

Dependências: R.JOO.MOBAPP.04.v1

Descrição:

A aplicação deverá verificar a conformidade do preenchimento do formulário, sendo que o utilizador deverá colocar informação em todos os campos do mesmo.

À semelhança do que se passa no passo 1, caso o utilizador não preencha o formulário de acordo com o estipulado deve aparecer uma mensagem em formato de alerta com o texto relativo ao problema.

Caso o utilizador não preencha nenhum campo deverá parecer uma mensagem com o texto "All fields are required!".

Caso o utilizador não preencha o campo "First Name" relativo ao seu primeiro nome deverá aparecer uma mensagem com o texto "First name is necessary!".

Caso o utilizador não preencha o campo "Last Name" relativo ao seu último nome deverá aparecer uma mensagem com o texto "Last name is necessary!".

Caso o utilizador não preencha o campo "City" relativo à sua cidade deverá aparecer uma mensagem com o texto "City is necessary!".

Caso o utilizador não escolha um valor no campo "Country" relativo ao seu país deverá aparecer uma mensagem com o texto "Country is necessary!".

Caso o utilizador não escolha um valor no campo "Industry" relativo à sua indústria deverá aparecer uma mensagem com o texto "Industry is necessary!".

Para fechar cada uma destas mensagens bastará ao utilizador clicar no botão de "OK".

4.6. Registrar na Aplicação – Passo 3

ID do requisito: R.JOO.MOBAPP.06.v1

Dependências: R.JOO.MOBAPP.04.v1

Descrição:

Ao clicar no botão “Next” do segundo passo do registo e caso o formulário seja corretamente validado o utilizador deve ser remetido para o terceiro passo do registo que deve ser apresentado de acordo com a Figura 6.

O utilizador deverá neste ecrã inserir as suas competências de forma individual. Devem ser requeridas pelo menos 5 competências e deverão ser aceites um máximo de 50.

Ao inserir texto no campo “Skill ou Expertise” deverá surgir um autocomplete relativo ao texto inserido com um máximo de 6 opções. O utilizador poderá escolher diretamente das opções apresentadas ficando essa competência escrita no campo e pronta a ser adicionada. O utilizador não poderá adicionar a mesma competência a sua lista de competências em duplicado. O utilizador poderá de igual forma inserir uma competência que o sistema ainda não conheça.

A qualquer altura o utilizador poderá retirar uma competência adicionada à lista clicando no x com fundo vermelho junto da mesma.

Na parte inferior da página devem aparecer 2 botões. Do lado esquerdo um botão com o texto “Back” permitirá ao utilizador regressar ao ponto anterior do registo. Do lado direito um botão com o texto “Next 2/3” permite ao utilizador saltar para o próximo passo do registo.

Figura 6 - Registo - Passo 3

4.7. Mensagens de Erro – Registo – Passo 3

ID do requisito: R.JOO.MOBAPP.07.v1

Dependências: R.JOO.MOBAPP.06.v1

Descrição:

À semelhança do que se passa no passo 1 e passo 2, caso o utilizador não preencha o formulário de acordo com o estipulado deve aparecer uma mensagem em formato de alerta com o texto relativo ao problema.

Caso o utilizador não adicione à sua lista pelo menos 5 competências deverá ser apresentada uma mensagem com o texto "Needed 5 skills!".

Caso o utilizador exceda as 50 skills deverá ser apresentada uma mensagem com o texto "You have reached the maximum number of skills!". Para fechar cada uma destas mensagens bastará ao utilizador clicar no botão de "OK".

Caso o utilizador remova um skill da lista deverá ser apresentado um alerta em que o título é o nome da skill que está a remover e com o texto "Remove skill from your profile?". Este alerta deverá apresentar dois botões, "Cancel" para fechar o alerta e "Ok" para fechar o alerta e remover a skill da lista.

4.8. Registrar na Aplicação – Passo 4

ID do requisito: R.JOO.MOBAPP.08.v1

Dependências: R.JOO.MOBAPP.06.v1

Descrição:

Quando o utilizador considera que as skills que introduziu são suficientes, garantindo o mínimo de quantidade e clica no botão “Next 2/3” deve ser remetido para o terceiro passo de registo onde fará

a escolha de alguns parâmetros de acordo com a Figura 7.

Nesses parâmetros será possível ao utilizador escolher para que países deseja receber/visualizar ofertas de emprego, se está interessado em receber notificações e se deve ser considerada a sua localização atual nas ofertas que lhe são sugeridas. Estas duas opções devem ser controladas através de um seletor de on/off permitindo ativar ou desativar a funcionalidade.

Na parte inferior da página devem aparecer 2 botões. Do lado esquerdo um botão com o texto “Back” permitirá ao utilizador regressar ao ponto anterior do registo. Do lado direito um botão com o texto “Finished” permite ao utilizador finalizar o registo enviando os dados para o servidor e completando o processo.

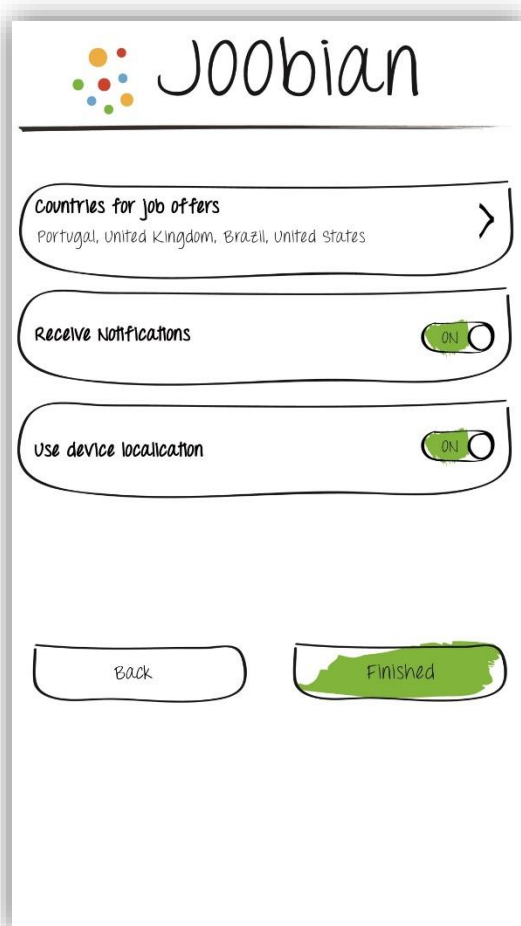


Figura 7 - Registo - Passo 4

4.9. Registrar na Aplicação – Passo 4 – Escolha de Países

ID do requisito: R.JOO.MOBAPP.09.v1

Dependências: R.JOO.MOBAPP.08.v1

Descrição:

Se o utilizador desejar escolher quais os países para os quais deseja receber/visualizar ofertas, deverá no ecrã de definições escolher a opção de escolha de países sendo remetido para um ecrã idêntico ao apresentado na Figura 8. Este ecrã deverá apresentar dinamicamente todos os países para os quais o servidor estiver preparado para receber ofertas.

A visibilidade das ofertas dos países deve ser controlada através de um seletor de on/off permitindo ativar ou desativar esses países. O utilizador deverá ter sempre pelo menos um país ativo e é informado disso através de uma mensagem no fim da lista de países com o texto "At least one country must be selected".

Na parte inferior da interface deverá estar disponível um botão com o texto "Back" que permita ao utilizador regressar ao ponto anterior, ou seja ao passo 4 do registo.

Durante todo o processo de registo a informação inserida pelo utilizador nas diversas etapas deve ser mantida quando este navega entre os passos.



Figura 8 - Registo - Passo 4 - Escolha de Países

4.10. Registrar na Aplicação via LinkedIn

ID do requisito: R.JOO.MOBAPP.010.v1

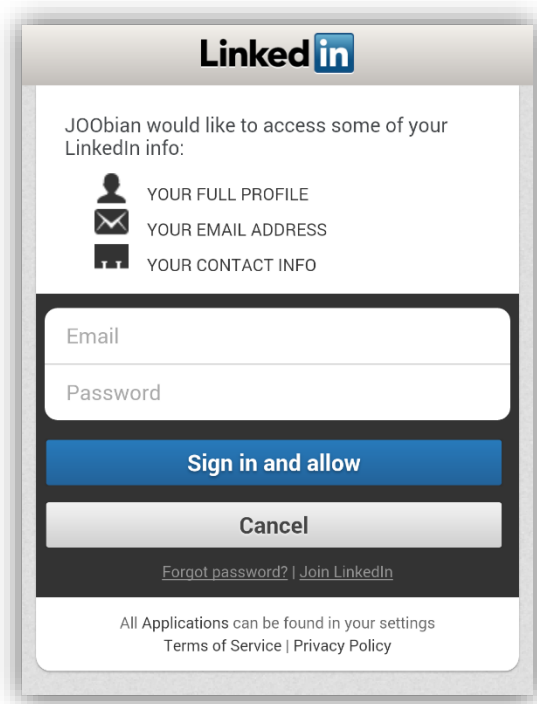
Dependências: R.JOO.MOBAPP.01.v1

Descrição:

Ao clicar no registo via LinkedIn deve ser apresentada ao utilizador uma mensagem, sob a forma de de alerta, com o título "Accept terms and conditions" e com a seguinte pergunta "I agree to the terms and conditions". Caso o utilizador concorde com os termos clicando em "OK", os quais pode aceder no ecrã do passo 1 do registo, deverá ser remetido para uma interface do LinkedIn de acordo com a Figura 9. Caso não concorde, clicando em "Cancel" a mensagem deverá desaparecer ficando no mesmo ecrã de registo.

Ao efetuar registo via LinkedIn o utilizador concorda que o JOObian acesse ao seu perfil em 3 perspetivas:

- Perfil completo
- Endereço de email
- Informação de contacto



Neste interface, totalmente da responsabilidade do LinkedIn o utilizador deverá inserir o seu email de registo nesta rede social bem como a sua password.

Em caso de falha nas credenciais utilizadas deverá aparecer uma mensagem da responsabilidade do LinkedIn consoante o erro detetado. Em caso de sucesso o utilizador deverá ser remetido para o Passo 2 do processo de registo em que aparece preenchido toda a informação possível de extrair do LinkedIn, prosseguindo depois para os restantes passos onde também deve estar preenchida a informação sobre skills caso o utilizador as tenha definido no seu perfil do LinkedIn.

Figura 9 - Registo LinkedIn

4.11. Autenticação na aplicação

ID do requisito: R.JOO.MOBAPP.011.v1

Dependências: R.JOO.MOBAPP.01.v1

Descrição:

Ao clicar na opção de “Sign In” no ecrã inicial o utilizador deve ser remetido para o ecrã onde pode escolher qual a forma de autenticação desejada de acordo com a Figura 10.

As duas opções que devem estar disponíveis são a autenticação via LinkedIn, que pode ser acedida

através de um botão no topo da interface onde deve estar o texto “Log in with LinkedIn”, ou a autenticação Joobian, na parte central da interface, que corresponde às credenciais criadas no processo de registo.

O formulário que permite a autenticação Joobian deverá ter um campo onde o utilizador deverá colocar o seu e-mail de registo e outro campo onde o utilizador coloca a sua password.

Na parte inferior da interface devem ser mostrados 2 botões. Do lado esquerdo um botão de “Cancel” que permite ao utilizador cancelar o processo de autenticação remetendo o mesmo para o ecrã inicial. Do lado direito um botão de “Done” que permite ao utilizador efetuar o processo de autenticação Joobian.

No fundo do ecrã deverá estar disponível um link com o texto “I forgot my password” que deverá permitir ao utilizador recuperar a sua password. Ao clicar nesse link o utilizador deverá ser remetido para uma interface com esse propósito.

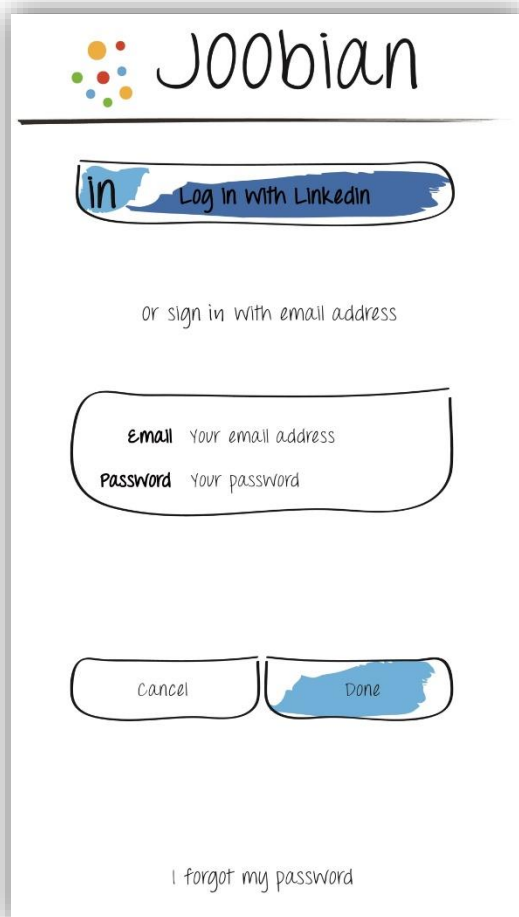


Figura 10 - Autenticação

4.12. Mensagens de Erro – Autenticação na aplicação

ID do requisito: R.JOO.MOBAPP.012.v1

Dependências: R.JOO.MOBAPP.011.v1

Descrição:

À semelhança do que acontece no processo de registo, caso o utilizador não preencha o formulário de acordo com o estipulado deve aparecer uma mensagem em formato de alerta com o texto relativo ao problema.

Caso o utilizador não tenha inserido nenhum texto no campo de email deve aparecer a mensagem com o texto "Email address needed!".

Caso o utilizador não tenha inserido nenhum texto no campo de password deve aparecer a mensagem com o texto "Password needed!".

Caso o utilizador tenha inserido um email não conforme deve aparecer uma mensagem com o texto "You have entered a invalid email address!".

Caso o utilizador tenha inserido um email não existentes na autenticação do Joobian deve aparecer uma mensagem com o texto "No account for this email. Please Sign Up for JOObian.", sendo o utilizador transferido para o primeiro passo do registo.

Para fechar cada uma destas mensagens bastará ao utilizador clicar no botão de "OK".

4.13. Autenticar na aplicação através do LinkedIn

ID do requisito: R.JOO.MOBAPP.013.v1

Dependências: R.JOO.MOBAPP.011.v1

Descrição:

Caso o utilizador se tenha registado via LinkedIn poderá depois de clicar na opção de "Sign In" no ecrã inicial selecionar a autenticação por este processo de acordo com a Figura 10.

O utilizador deverá nesta altura ser remetido para uma interface em tudo igual ao processo de registo, também esta de inteira responsabilidade do LinkedIn.

Em caso de falha no processo de autenticação esta interface será responsável pelas mensagens de erro de acordo com cada situação.

Em caso de sucesso o utilizador deverá ser remetido para interior da aplicação, neste caso para a listagem de ofertas de acordo com o seu perfil.

Caso o utilizador faça uso deste processo de autenticação, mas ainda não se tenha previamente registado na aplicação, então após a validação da autenticação por parte do LinkedIn o utilizador é remetido para o Passo 2 do processo de registo e tudo decorrerá de acordo com o requisito R.JOO.MOBAPP.010.v1.

4.14. Menu da aplicação

ID do requisito: R.JOO.MOBAPP.014.v1

Dependências: R.JOO.MOBAPP.011.v1

Descrição:

Depois de concluído o processo de autenticação e o utilizador ser remetido para o interior da aplicação, ou seja o seu core, deverá estar disponível transversalmente em todos os pontos da aplicação o menu na parte inferior do ecrã.

Este menu deverá ser constituído de forma ordenada pelos seguintes itens:

- Job Offers
- Favorites
- Profile
- Statistics
- Settings

Para ser possível situar em que ponto da aplicação o utilizador se encontra, o respetivo item do menu deverá aparecer de cor diferenciada.

O menu deverá funcionar de acordo com a Figura 11.

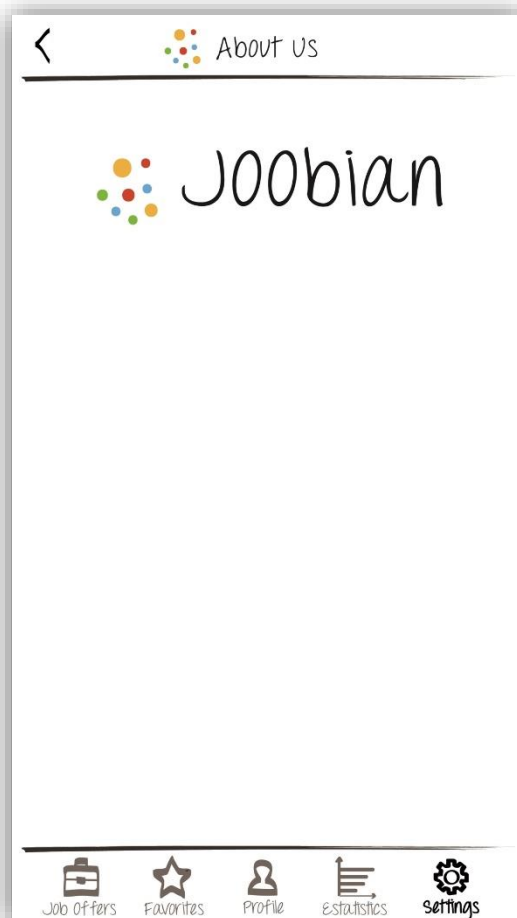


Figura 11 - Menu da aplicação

4.15. Listagem de ofertas de emprego

ID do requisito: R.JOO.MOBAPP.015.v1

Dependências: R.JOO.MOBAPP.014.v1

Descrição:

Após o utilizador se autenticar com sucesso na aplicação, será redirecionado para uma listagem de ofertas de empregos recomendadas para o seu perfil, segundo os seus Skills, Indústria e Settings definidas na aplicação de acordo com a Figura 12.

Esta listagem de ofertas de emprego deverá ter suporte para paginação, de forma a poder devolver as ofertas rapidamente, e deverá também haver um limite máximo de ofertas estipulado em 100. Em termos de paginação pretende-se receber grupos de 20 ofertas de cada vez e para tal o utilizador

deverá fazer scroll na lista de ofertas. Cada vez que for atingido o número de ofertas em memória deverá ser efetuado pedido de novo grupo, até ser atingido o limite de ofertas a mostrar.

Esta listagem poderá ser ordenada sendo que o utilizador poderá aceder à ordenação da listagem através do ícone colocado no topo direito da interface. Ao lado do ícone de ordenação também deverá estar disponível um ícone para efetuar pesquisas sobre as ofertas existentes para além desta listagem.

Cada oferta existente na listagem deverá apresentar na primeira linha a identificação da oferta, numa segunda linha a empresa responsável pela oferta e na terceira linha a localização da mesma. Lateralmente à direita deverá existir em cada oferta a indicação da sua antiguidade em horas, dias ou meses. Clicando em cada oferta deverá ser possível aceder aos seus detalhes.



Figura 12 - Lista de Ofertas de Emprego

4.16. Ordenação da Listagem da Oferta de Emprego

ID do requisito: R.JOO.MOBAPP.016.v1

Dependências: R.JOO.MOBAPP.015.v1

Descrição:

A listagem das ofertas de emprego poderá ser ordenada com base em três fatores, sendo eles a Relevância, a Data, e a Localização. Caso o utilizador altere este parâmetro o mesmo deverá ficar memorizado pela aplicação. O utilizador poderá aceder à ordenação da listagem através do ícone colocado no topo direito da interface de listagem de ofertas. Ao clicar neste ícone deverá aparecer uma mensagem idêntica à Figura 13, onde deverá estar selecionada a ordenação atual. O utilizador poderá escolher a mesma ou outra ordenação sendo que a listagem é renovada para refletir a escolha do utilizador. Caso o utilizador clique fora das opções disponíveis a mensagem deverá encerrar. No futuro poderá ser interessante que a listagem de ofertas possa também ser ordenada por outros fatores como por exemplo salário ou poupança salarial.

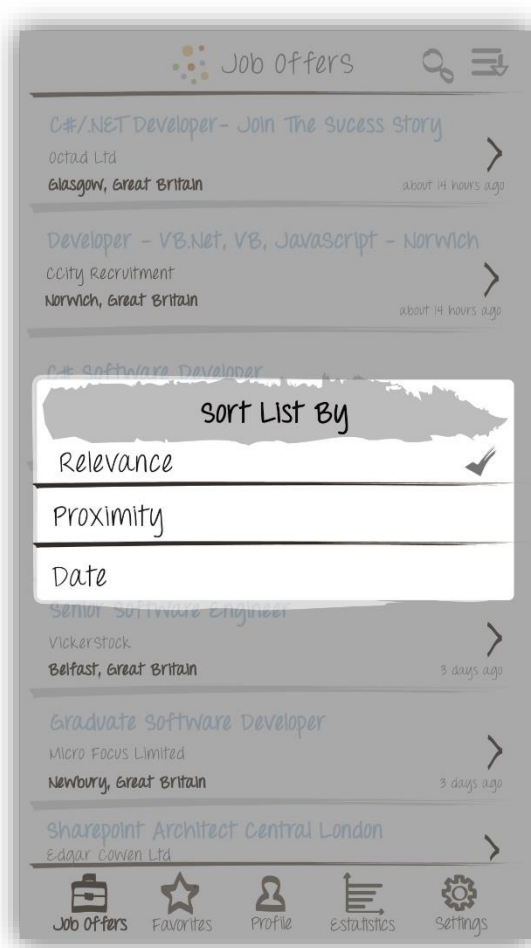


Figura 13 - Ordenação de Ofertas

4.17. Pesquisa de Ofertas de Emprego

ID do requisito: R.JOO.MOBAPP.017.v1

Dependências: R.JOO.MOBAPP.015.v1

Descrição:

Além das ofertas que são apresentadas ao utilizador deverá ser possível pesquisar por todas as ofertas disponíveis. Nesse sentido ao clicar sobre o ícone de pesquisa que se encontra no topo da interface da listagem de ofertas o utilizador deverá ser redirecionado para uma interface idêntica à Figura 14.

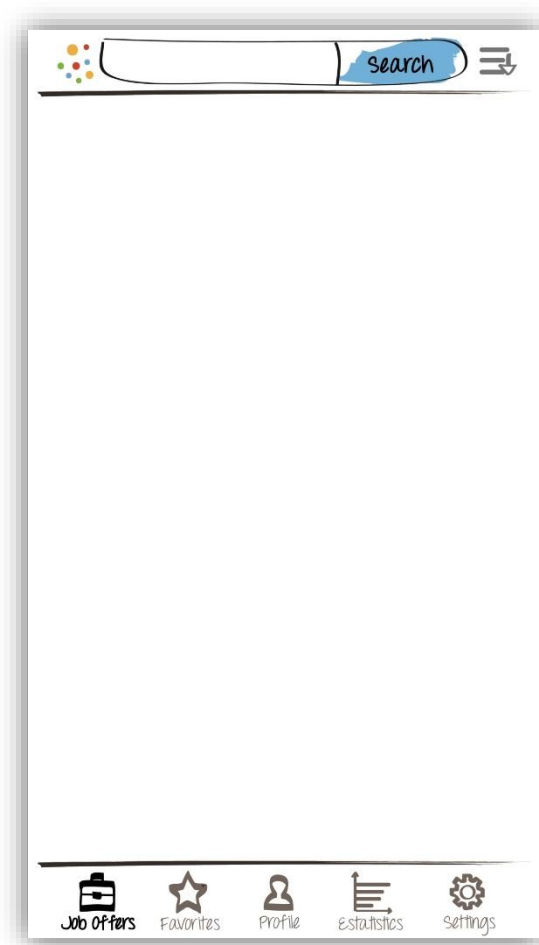


Figura 14 - Interface de Pesquisa

Na barra superior existe uma caixa de texto que permite pesquisar ofertas de emprego por skill, localização ou empresas. Para o efeito o utilizador deve inserir o termo a pesquisar e clicar em "Search". Do lado direito da caixa de pesquisa o utilizador pode de igual forma definir qual a ordenação pretendida para os resultados obtidos. Esta ordenação não deverá interferir com a ordenação da listagem normal de ofertas.

4.18. Resultados de Pesquisa

ID do requisito: R.JOO.MOBAPP.018.v1

Dependências: R.JOO.MOBAPP.017.v1

Descrição:

O utilizador ao inserir o termo de pesquisa e efetuar a mesma deverão ser apresentados os resultados de acordo com o termo pesquisado e ordenação estabelecida, devendo ser apresentados de forma idêntica ao da Figura 15.



Figura 15 - Resultados de Pesquisa

Os resultados de pesquisa devem ser divididos nas duas perspetivas "Recommended Offers" e "All Offers". Na primeira perspetiva o utilizador deverá visualizar as ofertas em que o termo de pesquisa seja encontrado e que de alguma forma se relacionem com o seu perfil. Na segunda perspetiva o utilizador deve visualizar todas as ofertas de acordo com o termo de pesquisa, mesmo que as mesmas não se relacionem com o seu perfil.

A visualização dos resultados deverá ser idêntica à listagem das ofertas inicial, permitindo de igual forma aceder aos detalhes de cada uma.

4.19. Detalhes da Oferta de Emprego

ID do requisito: R.JOO.MOBAPP.019.v1

Dependências: R.JOO.MOBAPP.015.v1

Descrição:

O utilizador ao carregar numa oferta de emprego e remetido para uma nova interface com a informação da mesma de acordo com a Figura 16. Nessa informação temos uma secção dedicada à identificação da oferta de emprego, empresa responsável, localização da oferta e antiguidade da mesma de forma idêntica ao que é apresentado na listagem.

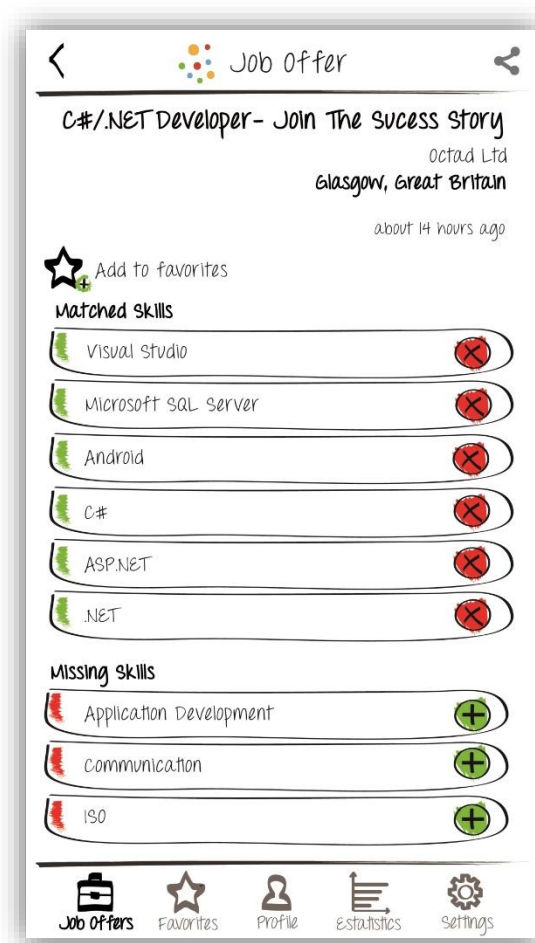


Figura 16 - Detalhes da Oferta de Emprego

Neste interface de detalhes o utilizador tem uma informação sobre:

- Favoritos
- Skills com Match
- Skills em falta
- Idiomas da oferta
- Descrição da oferta
- Responder à oferta.

4.20. Adicionar/Remover Oferta de Emprego aos Favoritos

ID do requisito: R.JOO.MOBAPP.020.v1

Dependências: R.JOO.MOBAPP.019.v1

Descrição:

Na interface de detalhes da oferta é mostrado um botão em forma de estrela de cor cinzenta para adicionar a oferta aos favoritos. Se o utilizador adicionar a oferta aos favoritos clicando na estrela, é mostrada uma "Toast" (mensagem informativa) a indicar que a oferta foi adicionada aos favoritos e a estrela muda de cor para amarelo.

Se a oferta já for favorita então a estrela já esta amarela. Se o utilizador remover a oferta dos favoritos clicando novamente na estrela, é mostrada uma "Toast" a indicar que a oferta foi removida dos favoritos e a estrela muda de cor para cinzento.

4.21. Matched Skills e Missing Skills

ID do requisito: R.JOO.MOBAPP.021.v1

Dependências: R.JOO.MOBAPP.019.v1

Descrição:

Na interface de detalhes da oferta deve ser mostrado um conjunto de skills que fazem match com o perfil do utilizador, ou seja competências identificadas na oferta que existem no perfil do utilizador e um conjunto de skills que foram identificadas na oferta mas que não constam do perfil do utilizador, ou seja competências que foi possível extrair da oferta mas que não fazem parte do grupo de competências que compõem o perfil do utilizador.

No caso das Matched Skills em cada uma delas o utilizador poderá clicar no ícone vermelho com o sinal de "X", para retirar essa skill do seu perfil. No caso das Missing Skills em cada uma delas o utilizador poderá clicar no ícone verde com o sinal de "+" para adicionar essa skill ao seu perfil.

Na primeira situação deve aparecer uma mensagem de confirmação com o nome da skill no topo e com o texto "Remove skill from your profile?". Em caso afirmativo a skill é removida do perfil. Na segunda situação deve aparecer uma mensagem de confirmação com o nome da skill no topo e com o texto "Add skill to your profile?". Em caso afirmativo a skill é adicionada ao perfil. Em ambas as situações ao cancelar a ação deve ser fechada a mensagem e nada mais deverá acontecer.

4.22. Idiomas da Oferta

ID do requisito: R.JOO.MOBAPP.022.v1

Dependências: R.JOO.MOBAPP.019.v1

Descrição:

Na interface de detalhes da oferta, após os Matched e Missing Skills, caso seja possível identificar da mesma informação sobre idiomas, deverá aparecer uma breve listagem com os idiomas identificados e que fazem parte dos requisitos da oferta.

4.23. Descrição da Oferta

ID do requisito: R.JOO.MOBAPP.023.v1

Dependências: R.JOO.MOBAPP.019.v1

Descrição:

Na interface de detalhes da oferta, após os idiomas, deverá ser apresentada toda a descrição da oferta, onde deverá estar todo o conteúdo que descreve a mesma e de onde foi possível extrair informação, nomeadamente as competências.

Ao clicar em cada uma das competências que existem no perfil do utilizador deverá ser assinalado a verde no texto da descrição onde está a mesma. De igual forma ao clicar sobre o nome da competência que foi identificada na oferta mas não existe no perfil deverá ser assinalado a vermelho onde a mesma se encontra no texto da descrição. Desta forma deverá ser mais simples ao utilizador verificar onde as mesmas se encontram no texto.

4.24. Ver oferta de emprego na fonte da oferta

ID do requisito: R.JOO.MOBAPP.024.v1

Dependências: R.JOO.MOBAPP.019.v1

Descrição:

Na interface de detalhes da oferta, após a descrição da oferta o utilizador deverá ter disponível uma imagem com o logótipo da fonte da oferta, ou seja a identificação de onde essa oferta foi recolhida. Ao clicar nessa imagem/botão o utilizador deve ser redirecionado através do browser do dispositivo para a página original da oferta.

4.25. Partilhar Oferta de Emprego

ID do requisito: R.JOO.MOBAPP.025.v1

Dependências: R.JOO.MOBAPP.019.v1

Descrição:

Na interface de visualização de detalhes de oferta no canto superior direito deverá existir um ícone que permita ao utilizador partilhar a oferta de emprego. A forma como pode ser efetuada a partilha é dinâmica e depende das aplicações que o utilizador tiver instaladas no seu dispositivo móvel.

Entre as diferentes possibilidades existentes deverá surgir a partilha por SMS ou Bluetooth, pelas diferentes contas de email configuradas no dispositivo, nas diferentes redes sociais tipo Facebook, LinkedIn, Google+, Twitter, poderão encontrar-se também diferentes aplicações de messaging como WhatsApp, Viber, Telegram, ou ainda aplicações como DropBox, Evernote, Feedly, QRCode, etc.

Estas possibilidades deverão parecer sobre a forma de um popup em que são listadas as diversas possibilidades de partilha. Depois de clicar em cada uma delas o comportamento depende inteiramente da aplicação escolhida.

A partilha é efetuada através do texto "*Offer JOObian - Job Search*" acrescido do título, empresa e localização da oferta. No final é adicionado um link para a origem da oferta sobre a forma de um link reduzido do tipo TinyUrl.

4.26. Favoritos

ID do requisito: R.JOO.MOBAPP.026.v1

Dependências: R.JOO.MOBAPP.014.v1

Descrição:

Ao clicar no ícone de “Favorites” do menu da aplicação o utilizador deverá ser remetido para a listagem de ofertas que considerou como sendo de interesse e tendo colocado na lista de favoritos. Essa listagem deverá existir de acordo com a Figura 17.

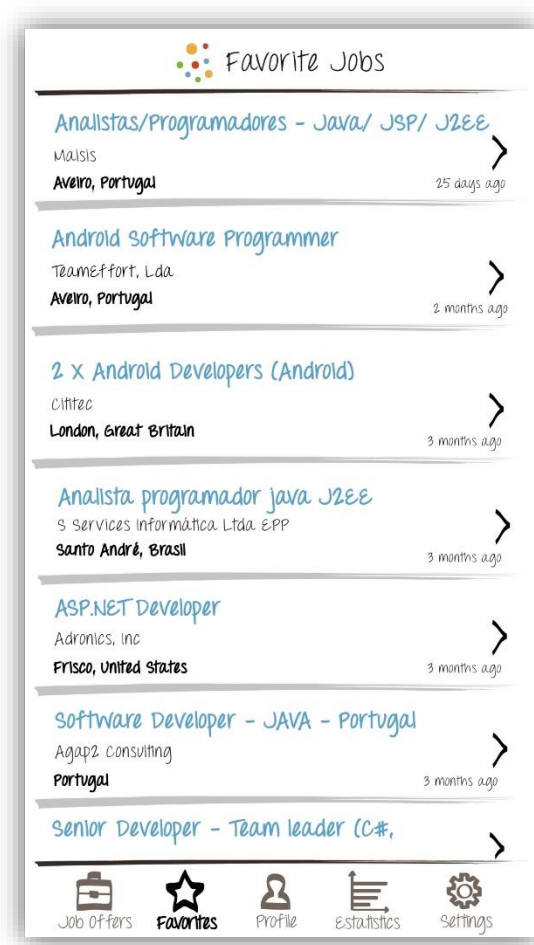


Figura 17 - Favoritos

Esta listagem deverá ser em tudo idêntica à listagem normal de ofertas, ou seja deve ser apresentado o título, a empresa e a localização por cada uma das ofertas, sendo que apenas são apresentadas as ofertas que o utilizador considerou como sendo as suas favoritas. Estas devem ser ordenadas pela data/hora em que foram inseridas nos favoritos. Ao clicar em cada uma das ofertas o utilizador deve de igual forma ser redirecionado para os detalhes da oferta.

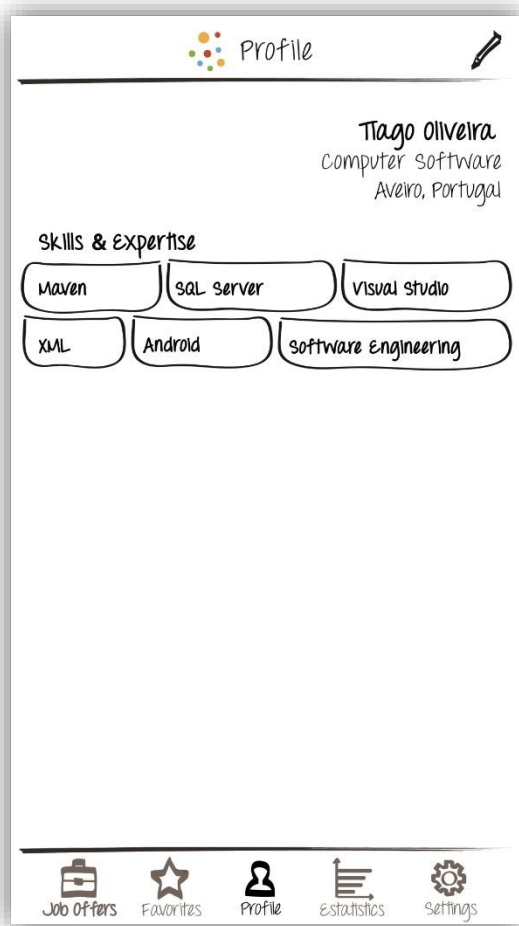
Caso o utilizador não efetue gestão desta lista de ofertas, ou seja não mantenha na lista apenas ofertas atuais, será normal que as ofertas mais antigas vão desaparecendo da lista de favoritos dado que deixam de estar disponíveis no servidor.

4.27. Visualização de Perfil do Utilizador

ID do requisito: R.JOO.MOBAPP.027.v1

Dependências: R.JOO.MOBAPP.014.v1

Descrição:



Ao clicar no ícone de “Profile” do menu da aplicação o utilizador deverá ser remetido para a interface de visualização do seu perfil, ou seja da visualização de informação que o caracteriza e que permite à aplicação ajustar as ofertas disponíveis. A interface de visualização deverá ser de acordo com a Figura 18.

Nesta interface será possível visualizar os dados que o utilizador definiu entre os quais, o seu nome, a sua indústria e a sua cidade. Deverá estar de igual forma visível todas as competências que o utilizador identificou como sendo adquiridas.

Além desta informação seria também de interesse que no perfil do utilizador pudessem constar a informação sobre as línguas que domina a sua experiência profissional e a sua educação ou percurso académico.

No canto superior desta interface deverá estar presente um ícone que permite ao utilizador editar a informação do seu perfil.

Figura 18 - Perfil do Utilizador

4.28. Edição de Perfil do Utilizador

ID do requisito: R.JOO.MOBAPP.028.v1

Dependências: R.JOO.MOBAPP.027.v1

Descrição:

Ao clicar no ícone de edição de perfil existente na interface de visualização do mesmo, o utilizador deverá ser redirecionado para uma interface onde pode atualizar a sua informação.

Nesta interface o utilizador deverá poder editar:

- Primeiro e Último Nome
- Cidade
- País
- Indústria
- Competências

Os campos do nome e cidade deverão ser campos abertos e os campos de país e indústria devem ser campos fechados onde o utilizador pode escolher entre os valores definidos pelo servidor Joobian. No campo de competências o utilizador deverá inserir as suas competências de forma individual. Devem ser requeridas pelo menos 5 competências e deverão ser aceites um máximo de 50.

Hand-drawn sketch of the 'Edit Profile' mobile app interface. The screen has a title bar with a logo, the text 'Edit Profile', and a save icon. Below the title bar are input fields for 'First Name' and 'Last Name', followed by a 'City' field. Then there are two dropdown menus labeled 'Choose Country' and 'Choose Industry', each with a checkmark icon. Below these is a 'Skill or expertise' field with a plus icon and the text '5 skills are required'. Underneath is a section titled 'Skills & Expertise' containing several skill tags: 'Maven', 'SQL server', 'Visual studio', 'XML', 'Android', 'Software Engineering', and 'Java'. Each tag has a red 'X' icon. At the bottom is a navigation bar with five icons: a briefcase for 'Job Offers', a star for 'Favorites', a person for 'Profile', a bar chart for 'Statistics', and a gear for 'Settings'.

Figura 19 - Edição de Perfil

Ao inserir texto no campo "Skill ou Expertise" deverá surgir um autocomplete relativo ao texto inserido com um máximo de 6 opções. O utilizador poderá escolher diretamente das opções apresentadas ficando essa competência escrita no campo e pronta a ser adicionada. O utilizador não poderá adicionar a mesma competência à sua lista de competências em duplicado. O utilizador poderá de igual forma inserir uma competência que o sistema ainda não conheça. A qualquer altura o utilizador poderá retirar uma competência existente na lista clicando no x com fundo vermelho junto da mesma. Caso o utilizador remova uma skill da lista deverá ser apresentado um alerta em que o título é o nome da skill que esta a remover e com o texto "Remove skill from your profile?". Este alerta deverá apresentar dois botões, "Cancel" para fechar o alerta e "Ok" para fechar o alerta e remover a skill da lista.

Para gravar as alterações efetuadas o utilizador deverá ter disponível no canto superior direito um ícone que lhe permite persistir o perfil. Quando clicar sobre esse ícone deverá aparecer uma mensagem com o texto "Profile saved successfully" indicando o sucesso da operação, bastando carregar no "OK" para fechar essa mensagem.

4.29. Estatística

ID do requisito: R.JOO.MOBAPP.029.v1

Dependências: R.JOO.MOBAPP.014.v1

Descrição:

Ao clicar no ícone de “Statistics” do menu da aplicação o utilizador deverá ser remetido para a interface de Estatística que deverá ser de acordo com a Figura 20.

Esta interface deverá apresentar no topo do lado direito um resumo da informação analisada, ou seja qual o número de ofertas que foram analisadas e qual o número de ofertas disponíveis para o perfil

do utilizador em função do país ou países para os quais o utilizador deseja receber ofertas.

As estatísticas devem ser divididas em 4 vetores em função dos parâmetros atrás mencionados:

- As competências mais relevantes (Most Relevant Skills)
- As competências que devem ser adquiridas (Must Have Skills)
- As localizações com mais ofertas (Hot Locations)
- As empresas com mais ofertas (Hot Companies)

Em cada um destes vetores são considerados até um máximo de 13 resultados relevantes sendo que os números apresentados em cada ponto dizem respeito ao número de ofertas em que o mesmo é mencionado. Em cada ponto as barras dizem respeito a uma percentagem relativa ao nº de ofertas analisadas.

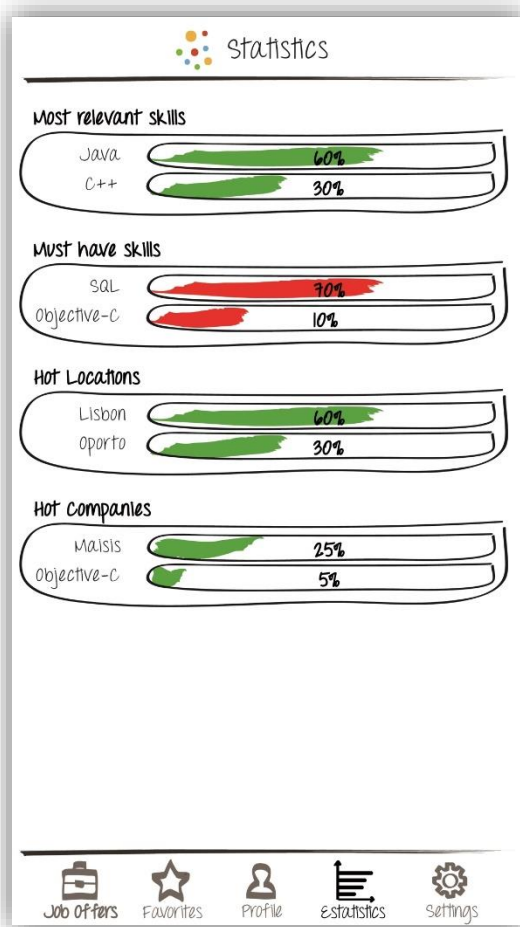


Figura 20 - Estatística

4.30. Settings

ID do requisito: R.JOO.MOBAPP.030.v1

Dependências: R.JOO.MOBAPP.014.v1

Descrição:

Ao clicar no ícone de “Settings” do menu da aplicação o utilizador deverá ser remetido para a interface de definições da aplicação. Essa interface deverá permitir ao utilizador definir aspetos transversais ao uso da aplicação e deverá ser de acordo com a Figura 21.

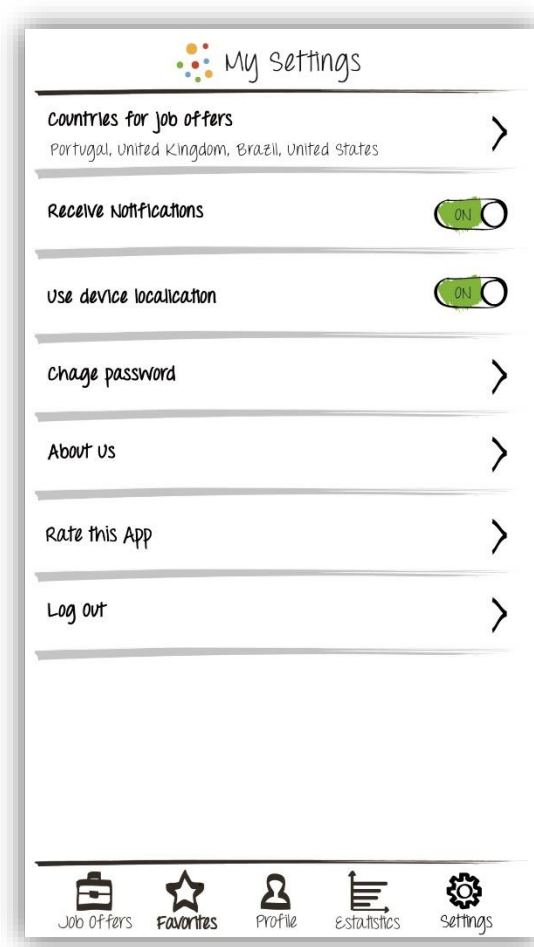


Figura 21 - Settings

Entre as definições e ações disponíveis devem estar:

- Escolha dos países das ofertas
- Receber notificações
- Utilizar a localização atual
- Mudar a palavra-passe (apenas no caso de o utilizador estar na aplicação através da autenticação Joobian)
- Sobre a aplicação
- Avaliar a aplicação na loja de aplicações
- Sair da aplicação

4.31. Escolha de Países

ID do requisito: R.JOO.MOBAPP.031.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Entre as possíveis definições da aplicação, a escolha de países para os quais o utilizador deseja receber ofertas será certamente uma das mais importantes. Ao aceder à interface de “Settings” o utilizador deverá visualizar numa listagem separada por virgulas, de quais os países que tem selecionado e para os quais são listadas ofertas e são analisadas essas mesmas ofertas para efeitos de estatística de acordo com a Figura 21.

Ao clicar sobre essa lista o utilizador deve ser redirecionado para uma interface de acordo com a Figura 22, onde pode escolher individualmente quais os países para os quais pretende receber ofertas. Essa definição é efetuada através de um selecionador ON/OFF sendo as escolhidas aquelas que estão ON, ou seja a verde.

A listagem de países é mantida pelo servidor e à medida que forem adicionados novos países à recolha de ofertas, isso deverá ser refletido nesta listagem em termos dos clientes móveis.



Figura 22 - Escolha de Países

4.32. Receber notificações

ID do requisito: R.JOO.MOBAPP.032.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Na interface de “Settings” da Figura 21, outro dos parâmetros que o utilizador pode definir é se pretende receber notificações nativas do sistema operativo do telemóvel. Ao ativar esta opção o utilizador irá receber numa base diária, quantas novas ofertas foram agregadas pelo Joobian desde a última notificação ou utilização da aplicação por parte do utilizador, contando sempre aquela que ocorreu há menos tempo.

O utilizador deverá então, caso ative as notificações, receber as mesmas com o texto “You have X new job offers.”, onde o X representa o número de novas ofertas até um máximo de 50. Caso o utilizador tenha mais de 50 novas ofertas é indicado 50+ no lugar do X.

Ao clicar sobre a notificação o utilizador deverá ser remetido para a aplicação, podendo ou não ser requerido login dependendo se o utilizador tenha efetuado o logout na última utilização da aplicação.

4.33. Utilizar a localização atual

ID do requisito: R.JOO.MOBAPP.033.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Na interface de “Settings” da Figura 21, outro dos parâmetros que o utilizador pode definir é se pretende utilizar a localização onde se encontra como elemento na avaliação das ofertas recomendadas para o seu perfil.

Caso esta seleção esteja ativa, quando o utilizador pretender listar as ofertas recomendadas por proximidade a sua localização atual será tida em conta, caso contrário a localização que será tida em conta é aquela que foi definida no perfil.

Desta forma o utilizador poderá mais facilmente encontrar ofertas perto do local em que se encontra.

4.34. Mudar palavra-passe

ID do requisito: R.JOO.MOBAPP.034.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Na interface de “Settings” da Figura 21 uma das ações disponíveis é a alteração da palavra-passe através do botão “Change password”. Esta ação só deverá estar disponível para os utilizadores que usem a aplicação com a autenticação Joobian, ou seja, os utilizadores que entrem via LinkedIn não devem ver esta opção.

Ao clicar nesta ação o utilizador deve ser remetido para uma interface de acordo com a Figura 23.

Nesta interface o utilizador deverá ter 3 campos onde insira a palavra-passe atual, a nova palavra-

passe e o terceiro campo para inserir a confirmação da nova palavra-passe. A nova palavra-passe deverá respeitar o tamanho mínimo de 6 caracteres.

Quando o utilizador clica no botão de “Change Password” para efetuar a operação, é validado o correto preenchimento dos diferentes campos. Caso o utilizador não introduza a palavra-passe atual correta deverá ser mostrada uma mensagem com o texto “Old password is not the same”. Caso a nova palavra passe tenha menos do que os 6 caracteres necessários, deverá parecer uma mensagem com o texto “Your password has not 6 characters!”. Caso a palavra-passe inserida no campo de nova palavra-passe e aquela inserida na confirmação da palavra-passe sejam diferentes deverá aparecer uma mensagem com o texto “The passwords are not the same!”. Em caso de sucesso deverá parecer uma mensagem com o texto “Your password changed successfully”.

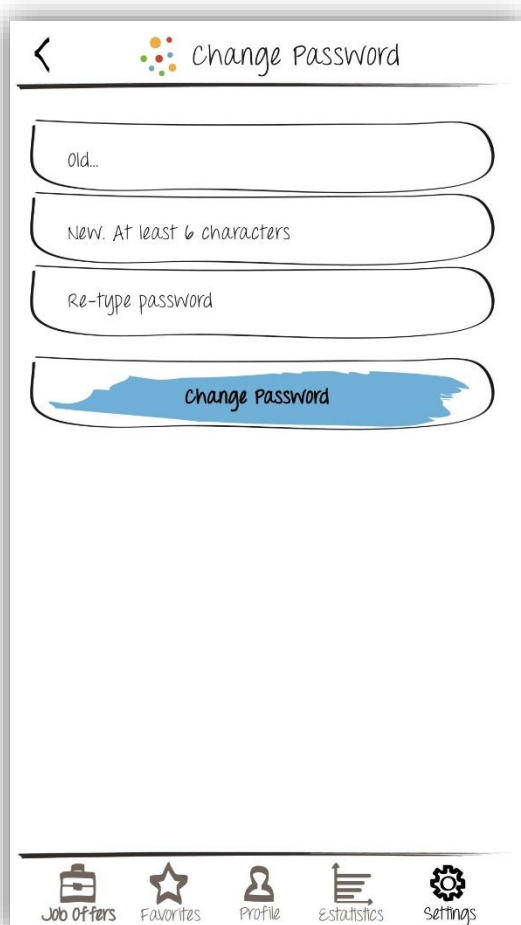


Figura 23 - Alteração de palavra-passe

4.35. Sobre a aplicação

ID do requisito: R.JOO.MOBAPP.035.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Na interface de “Settings” da Figura 21 uma das ações disponíveis é a possibilidade de ver informação sobre a aplicação através do botão “About us”. Ao clicar neste botão o utilizador deve ser redirecionado para uma interface de acordo com a Figura 24, onde é possível ver a versão da aplicação que o utilizador está a utilizar e uma descrição sobre a aplicação. No final dessa descrição deverá existir um link para o site do Joobian.



Figura 24 - Sobre a aplicação

4.36. Avaliar a aplicação

ID do requisito: R.JOO.MOBAPP.036.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Na interface de “Settings” da Figura 21 uma das ações disponíveis é a possibilidade de avaliar a aplicação Joobian na loja de aplicações do sistema operativo do dispositivo móvel através do botão “Rate this app”.

Ao clicar neste botão o utilizador deve ser redirecionado para a página da aplicação em cada uma das lojas permitindo dessa forma avaliar a aplicação.

4.37. Sair da aplicação

ID do requisito: R.JOO.MOBAPP.037.v1

Dependências: R.JOO.MOBAPP.030.v1

Descrição:

Na interface de “Settings” da Figura 21 uma das ações disponíveis é a possibilidade de sair da aplicação através do botão “Logout”.

Ao clicar neste botão é encerrada a sessão do utilizador redirecionando o mesmo para o ecrã inicial da aplicação.

4.38. Termos de Serviço e Política de Privacidade

ID do requisito: R.JOO.MOBAPP.038.v1

Dependências: R.JOO.MOBAPP.02.v1

Descrição:

Na interface de registo da aplicação de acordo com a Figura 4 deverá aparecer na parte inferior um link com o texto "Terms of Service & Privacy Policy". Ao clicar neste link o utilizador deve ser redirecionado para uma interface onde será possível ver todos os termos de serviço e política de privacidade, de acordo com a Figura 25. O texto presente na aplicação pode ser lido na íntegra no ponto 6.1 do presente documento.

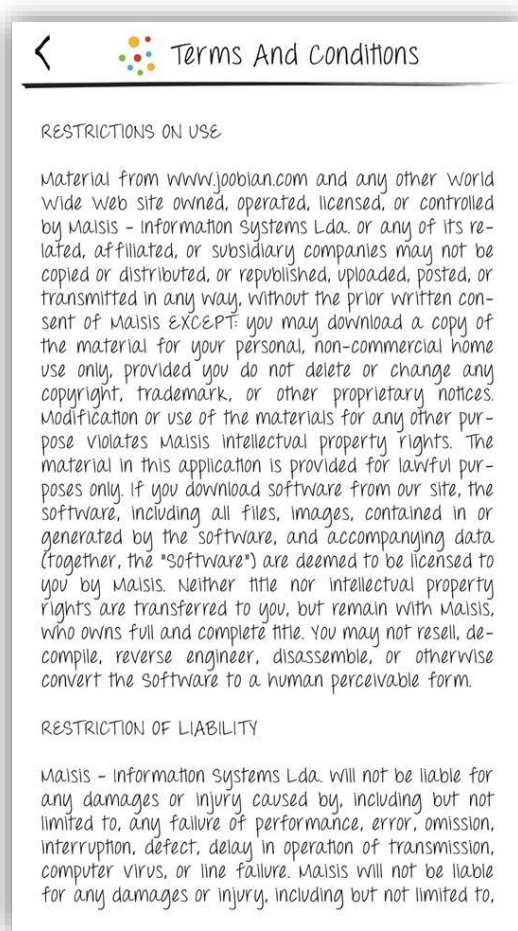


Figura 25 - Termos de Serviço e Política de Privacidade

4.39. Esqueci Password

ID do requisito: R.JOO.MOBAPP.039.v1

Dependências: R.JOO.MOBAPP.011.v1

Descrição:

Na interface de autenticação na aplicação de acordo com a Figura 10 deverá estar disponível um link com o texto "I forgot my password" que deverá permitir ao utilizador recuperar a sua password. Ao clicar nesse link o utilizador deverá ser remetido para uma interface com esse propósito de acordo com a Figura 26.

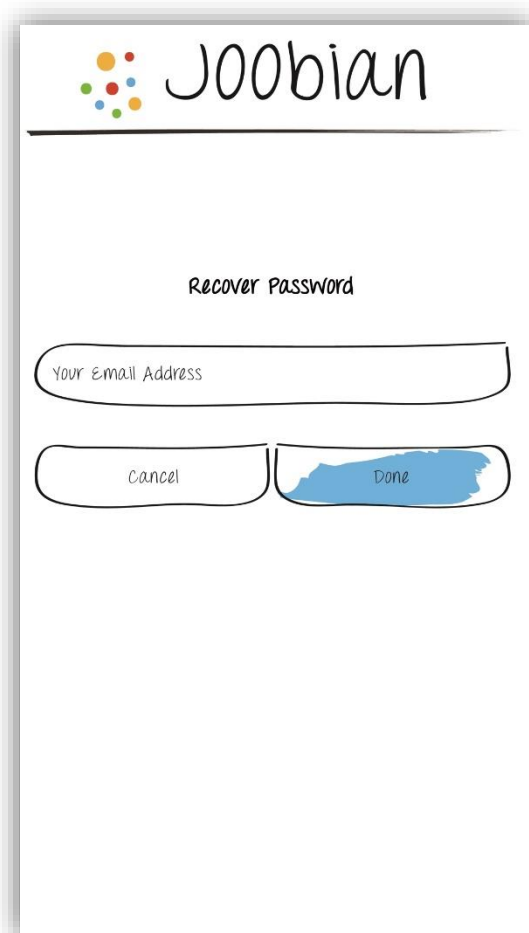


Figura 26 - Recuperação de Password

Nesta interface o utilizador deverá ter disponível um campo para inserir o endereço de correio eletrónico utilizado no registo Joobian. Ao clicar no botão "Done" é efetuada a validação do email introduzido. Caso o formato do email inserido seja inválido deverá aparecer uma mensagem com o texto "You have entered an invalid email address!". Caso o email inserido não exista no registo do Joobian deverá aparecer uma mensagem com o texto "No account for this email. Please Sign Up for JOObian.". Em caso de sucesso deverá ser apresentada uma mensagem com o texto "New password was sent to your email.". Neste caso o utilizador deverá receber no seu correio eletrónico uma nova palavra-passe gerada aleatoriamente que poderá usar de imediato para aceder à aplicação.

4.40. Hardware do dispositivo

ID do requisito: R.JOO.MOBAPP.040.v1

Dependências:

Descrição:

A aplicação deverá tirar partido do hardware do dispositivo. No caso do GPS, de forma a poder obter a localização do utilizador, no caso do WiFi da ligação de dados e a localização, de forma a poder obter dados e a localização do utilizador e no caso da ligação de dados (3G/4G), de forma a poder obter dados.

Os dados são críticos para a utilização da aplicação pois sem dados não é possível ir buscar novas ofertas ou até mesmo autenticar na aplicação. Quanto à localização esta será usada para dar resultados de ofertas de emprego perto do utilizador.

4.41. Requisitos de performance

ID do requisito: R.JOO.MOBAPP.041.v1

Dependências:

Descrição:

A aplicação deverá ser o mais leve e responsiva possível, por leve pretende-se que a aplicação tenha um tamanho de pacote inferior a 10Mb, consumo de memória RAM reduzido (abaixo dos 100Mb), uso de CPU reduzido e uso de dados reduzido.

4.42. Recuperação do sistema em caso de falha

ID do requisito: R.JOO.MOBAPP.042.v1

Dependências:

Descrição:

A aplicação deverá poder continuar a ver as ofertas de emprego mesmo na eventualidade da quebra do servidor ou da ligação de dados do telemóvel/dispositivo.

4.43. Ver oferta salarial

ID do requisito: R.JOO.MOBAPP.043.v1

Dependências: R.JOO.MOBAPP.023.v1

Descrição:

Na interface de detalhes da oferta além da informação já descrita sobre a mesma, poderá ser útil disponibilizar a informação sobre o valor salarial da oferta. Apenas em algumas das ofertas que são recolhidas pelo servidor do Joobian essa informação está disponível. No entanto para uma análise mais cuidada sobre a oferta essa informação pode ser crucial. Essa informação deverá ser disponibilizada na moeda de origem da oferta que poderá ser diferente da moeda do local definido pelo perfil do utilizador.

4.44. Ver Index salário/custo de vida

ID do requisito: R.JOO.MOBAPP.044.v1

Dependências: R.JOO.MOBAPP.023.v1

Descrição:

Na interface de detalhes da oferta além da informação já descrita sobre a mesma, poderá ser útil disponibilizar a informação sobre o índice da oferta salarial versus o custo de vida médio na localização da oferta.

Existem fontes de dados públicos passíveis de serem consultadas onde estão disponíveis os valores atualizados em relação ao custo médio de vida em diferentes localizações. Apenas em algumas das ofertas que são recolhidas pelo servidor do Joobian a informação sobre o salário está disponível. No entanto para uma análise mais cuidada sobre a oferta este índice pode ser crucial.

4.45. Importar CV

ID do requisito: R.JOO.MOBAPP.045.v1

Dependências: R.JOO.MOBAPP.028.v1

Descrição:

Na interface do perfil do utilizador onde é possível ver a informação do utilizador que foi inserida ou editada pelo mesmo, poderia existir forma de efetuar o upload de um CV para o servidor Joobian e dessa forma extrair conteúdo que pudesse complementar a informação já inserida de uma forma automática sem que o utilizador necessitasse de inserir através da interface do dispositivo móvel toda a informação que compõe o seu perfil.

5. Listagem de Requisitos

Código	Nome	Versão	Prioridade	Dependências	Data				Observações
					Apresentação	Confirmação	Alteração	Eliminação	
R.JOO.MOBAPP.01.v1	Ecrã Inicial	1.0	Normal		2014-06-11	2014-06-11			
R.JOO.MOBAPP.02.v1	Registar na Aplicação – Passo 1	1.0	Normal	R.JOO.MOBAPP.01.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.03.v1	Mensagens de Erro – Registo – Passo 1	1.0	Normal	R.JOO.MOBAPP.02.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.04.v1	Registar na Aplicação – Passo 2	1.0	Normal	R.JOO.MOBAPP.02.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.05.v1	Mensagens de Erro – Registo – Passo 2	1.0	Normal	R.JOO.MOBAPP.04.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.06.v1	Registar na Aplicação – Passo 3	1.0	Normal	R.JOO.MOBAPP.04.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.07.v1	Mensagens de Erro – Registo – Passo 3	1.0	Normal	R.JOO.MOBAPP.06.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.08.v1	Registar na Aplicação – Passo 4	1.0	Normal	R.JOO.MOBAPP.06.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.09.v1	Registar na Aplicação – Passo 4 – Escolha de Países	1.0	Normal	R.JOO.MOBAPP.08.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.10.v1	Registar na Aplicação via LinkedIn	1.0	Normal	R.JOO.MOBAPP.08.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.11.v1	Autenticação na aplicação	1.0	Normal	R.JOO.MOBAPP.01.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.12.v1	Mensagens de Erro – Autenticação na aplicação	1.0	Normal	R.JOO.MOBAPP.011.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.13.v1	Autenticar na aplicação através do LinkedIn	1.0	Normal	R.JOO.MOBAPP.011.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.14.v1	Menu da aplicação	1.0	Normal	R.JOO.MOBAPP.011.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.15.v1	Listagem de ofertas de emprego	1.0	Normal	R.JOO.MOBAPP.014.v1	2014-06-11	2014-06-11			

R.JOO.MOBAPP.16.v1	Ordenação da Listagem da Oferta de Emprego	1.0	Normal	R.JOO.MOBAPP.015.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.17.v1	Pesquisa de Ofertas de Emprego	1.0	Normal	R.JOO.MOBAPP.015.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.18.v1	Resultados de Pesquisa	1.0	Normal	R.JOO.MOBAPP.015.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.19.v1	Detalhes da Oferta de Emprego	1.0	Normal	R.JOO.MOBAPP.015.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.20.v1	Adicionar/Remover Oferta de Emprego aos Favoritos	1.0	Normal	R.JOO.MOBAPP.019.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.21.v1	Matched Skills e Missing Skills	1.0	Normal	R.JOO.MOBAPP.019.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.22.v1	Idiomas da Oferta	1.0	Normal	R.JOO.MOBAPP.019.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.23.v1	Descrição da Oferta	1.0	Normal	R.JOO.MOBAPP.019.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.24.v1	Ver oferta de emprego na fonte da oferta	1.0	Normal	R.JOO.MOBAPP.019.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.25.v1	Partilhar Oferta de Emprego	1.0	Normal	R.JOO.MOBAPP.019.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.26.v1	Favoritos	1.0	Normal	R.JOO.MOBAPP.014.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.27.v1	Visualização de Perfil do Utilizador	1.0	Normal	R.JOO.MOBAPP.014.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.28.v1	Edição de Perfil do Utilizador	1.0	Normal	R.JOO.MOBAPP.27.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.29.v1	Estatística	1.0	Normal	R.JOO.MOBAPP.014.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.30.v1	Settings	1.0	Normal	R.JOO.MOBAPP.014.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.31.v1	Escolha de Países	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.32.v1	Receber notificações	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.33.v1	Utilizar a localização atual	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			

R.JOO.MOBAPP.34.v1	Mudar palavra-passe	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.35.v1	Sobre a aplicação	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.36.v1	Avaliar a aplicação	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.37.v1	Sair da aplicação	1.0	Normal	R.JOO.MOBAPP.030.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.38.v1	Termos de Serviço e Política de Privacidade	1.0	Normal	R.JOO.MOBAPP.02.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.39.v1	Esqueci Password	1.0	Normal	R.JOO.MOBAPP.011.v1	2014-06-11	2014-06-11			
R.JOO.MOBAPP.40.v1	Hardware do dispositivo	1.0	Normal		2014-06-11	2014-06-11			
R.JOO.MOBAPP.41.v1	Requisitos de performance	1.0	Normal		2014-06-11	2014-06-11			
R.JOO.MOBAPP.42.v1	Recuperação do sistema em caso de falha	1.0	Normal		2014-06-11	2014-06-11			
R.JOO.MOBAPP.43.v1	Ver oferta salarial	1.0	Normal	R.JOO.MOBAPP.023.v1	2014-06-11	2014-06-11			Later
R.JOO.MOBAPP.44.v1	Ver Index salário/custo de vida	1.0	Normal	R.JOO.MOBAPP.023.v1	2014-06-11	2014-06-11			Later
R.JOO.MOBAPP.45.v1	Importar CV	1.0	Normal	R.JOO.MOBAPP.028.v1	2014-06-11	2014-06-11			Later

6. Anexos

6.1. Termos de Serviço e Política de Privacidade

RESTRICTIONS ON USE

Material from www.joobian.com and any other World Wide Web site owned, operated, licensed, or controlled by Maisis – Information Systems Lda. or any of its related, affiliated, or subsidiary companies may not be copied or distributed, or republished, uploaded, posted, or transmitted in any way, without the prior written consent of Maisis EXCEPT: you may download a copy of the material for your personal, non-commercial home use only, provided you do not delete or change any copyright, trademark, or other proprietary notices. Modification or use of the materials for any other purpose violates Maisis intellectual property rights. The material in this application is provided for lawful purposes only. If you download software from our site, the software, including all files, images, contained in or generated by the software, and accompanying data (together, the "Software") are deemed to be licensed to you by Maisis. Neither title nor intellectual property rights are transferred to you, but remain with Maisis, who owns full and complete title. You may not resell, decompile, reverse engineer, disassemble, or otherwise convert the Software to a human perceivable form.

RESTRICTION OF LIABILITY

Maisis – Information Systems Lda. will not be liable for any damages or injury caused by, including but not limited to, any failure of performance, error, omission, interruption, defect, delay in operation of transmission, computer virus, or line failure. Maisis will not be liable for any damages or injury, including but not limited to, special or consequential damages that result from the use of, or the inability to use, the materials in this application, even if there is negligence or Maisis or an authorized Maisis representative has been advised of the possibility of such damages, or both. The above limitation or exclusion may not apply to you to the extent that applicable law may not allow the limitation or exclusion of liability for incidental or consequential damages. Maisis total liability to you for all losses, damages, and causes of action (in contract, tort (including without limitation, negligence), or otherwise) will not be greater than the amount you paid to access this application.

SUBMISSIONS

All remarks, suggestions, ideas, graphics, or other information communicated to Maisis – Information Systems Lda. through this application (together, the "Submission") will forever be the property of Maisis. Maisis will not be required to treat any Submission as confidential, and will not be liable for any ideas for its business (including without limitation, product, or advertising ideas) and will not incur any liability as a result of any similarities that may appear in future Maisis operations. Without limitation, Maisis will have exclusive ownership of all present and future existing rights to the Submission of every kind and nature everywhere. Except as noted below in this paragraph, Maisis

will be entitled to use the Submission for any commercial or other purpose whatsoever without compensation to you or any other person sending the Submission. Personally identifiable information that may be received at this application is provided voluntarily by a visitor to this application. This information is for internal purposes only and is not sold or otherwise transferred to third parties of Maisis or to other entities who are not involved in the operation of this application. Information submitted via a number of areas in this application is not retained. Therefore, the above right to use Submissions is subject to this limited use of this information and excludes non-retained information. You acknowledge that you are responsible for whatever material you submit, and you, not Maisis have full responsibility for the message, including its legality, reliability, appropriateness, originality, and copyright.

PRIVACY POLICY

What information do we collect?

We collect information from you that is provided by LinkedIn® when you login on our mobile application.

When logging on our application, as appropriate, you may be asked to enter your LinkedIn® credentials in order to obtain your public profile on LinkedIn® and also your personal skills.

What do we use your information for?

Any of the information we collect from you may be used in one of the following ways:

- To personalize your experience (your information helps us to better respond to your individual needs)
- To improve our mobile application (we continually strive to improve our mobile application offerings based on the information and feedback we receive from you)
- To improve customer service (your information helps us to more effectively respond to your customer service requests and support needs)
- To process transactions

Your information, whether public or private, will not be sold, exchanged, transferred, or given to any other company for any reason whatsoever, without your consent, other than for the express purpose of delivering the purchased product or service requested by the customer.

- To send periodic emails

The email address you provide, may be used to send you information and updates pertaining to your order, in addition to receiving occasional company news, updates, related product or service information, etc.

Note: If at any time you would like to unsubscribe from receiving future emails, we include detailed unsubscribe instructions at the bottom of each email.

- To administer a contest, promotion, survey or other site feature

How do we protect your information?

We implement a variety of security measures to maintain the safety of your personal information when you access your personal information.

Do we use cookies?

We do not use cookies.

Do we disclose any information to outside parties?

We do not sell, trade, or otherwise transfer to outside parties your personally identifiable information. This does not include trusted third parties who assist us in operating our mobile application, conducting our business, or servicing you, so long as those parties agree to keep this information confidential. We may also release your information when we believe release is appropriate to comply with the law, enforce our site policies, or protect ours or others rights, property, or safety. However, non-personally identifiable visitor information may be provided to other parties for marketing, advertising, or other uses.

Third party links

Occasionally, at our discretion, we may include or offer third party products or services on our mobile application. These third party sites or mobile applications have separate and independent privacy policies. We therefore have no responsibility or liability for the content and activities of these linked sites. Nonetheless, we seek to protect the integrity of our mobile application and welcome any feedback about these sites.

Online Privacy Policy Only

This online privacy policy applies only to information collected through our mobile application and not to information collected online.

Your Consent

By using our mobile application, you consent to our privacy policy.

Changes to our Privacy Policy

If we decide to change our privacy policy, we will post those changes on this page.

This policy was last modified on 2013/11/20

Contacting Us

If there are any questions regarding this privacy policy you may contact us using the information on the URL below.

<http://www.joobian.com>

Alertamos para o facto das cópias em papel poderem estar desatualizadas.

Consulte o uebe.Q para obter a última versão.

Apêndice B

Documento de Testes

JOObian

Aplicação Móvel

Plano de Testes

23/06/2014

v5.0

Rui Macedo

Histórico de versões do Plano de Testes

Versão	Datas	Responsabilidades	Justificação
5.0	23/06/2014	Edição: Rui Macedo	JOObian (BETA).
	23/06/2014	Revisão: Rui Macedo	
	23/06/2014	Aprovação: Rui Macedo	

Índice

1.	Introdução	4
1.1.	Documentos Relacionados.....	4
1.2.	Estrutura do Documento	4
2.	Avaliação dos Casos de Teste.....	5
3.	Descrição do Sistema	6
3.1.	Motivação e Objetivos.....	6
3.2.	Público-alvo	7
3.3.	Plataforma JOOBIAN.....	8
4.	Casos de Teste	11
4.1.	Acesso	11
	Sign in	11
4.2.	Settings.....	12
	Countries for Jobs Offers	12

1. Introdução

Este documento é o Plano de Testes para a aplicação JOObian - BETA.

1.1. Documentos Relacionados

Documentos	Breve Descrição
<i>REQ_JOO_v5.0_JOObian_MobileApp.doc</i>	Este documento descreve os requisitos da aplicação JOObian (BETA).

1.2. Estrutura do Documento

Serão descritos todos os casos de teste relativos à aplicação JOObian (BETA).

2. Avaliação dos Casos de Teste

ID do Caso de Teste	Título do Teste	Passou	Falhou	Observações
JOOB.A_SignIn.01v1	Sign in			
JOOB.S_CJO.01v1	Countries for Jobs Offers			

3. Descrição do Sistema

A aplicação móvel Joobian pretende efetuar o cruzamento das ofertas de emprego existentes no mercado com o perfil de cada utilizador, possibilitando dessa forma que este último obtenha as ofertas de trabalho que melhor se adequam ao seu perfil.

O cruzamento efetuado tem como parâmetros os diversos pontos do perfil do utilizador e as diferentes características da oferta. De uma forma inicial tudo pode ser efetuado de forma semiautomática ao seja com a interligação com o LinkedIn onde o utilizador já definiu o seu perfil e com a análise ao texto da oferta esse cruzamento é possível. No entanto para que a garantia de bons resultados seja cada vez maior o utilizador poderá alterar parâmetros no seu perfil, tentando que as ofertas sejam mais adequadas ao mesmo.

3.1. Motivação e Objetivos

A plataforma de emprego Joobian será construída com o intuito de ter o maior número de dados tanto de perfis de candidatos a emprego como de ofertas de emprego pois, quantos mais dados houver na plataforma, mais valor será possível gerar através desta. Tendo este como o principal objetivo, é necessário que a aplicação/cliente móvel aqui detalhado maximize esta estratégia, para isso é vital cumprir com os seguintes pontos-chave:

- Facilitar e incentivar que mais pessoas se juntem à plataforma e registem os seus dados – Assim será possível fazer com que mais pessoas (e empresas) adiram à plataforma e registem nela os seus dados;
- Possibilidade de integração com sistemas já existentes – Havendo integração com outros sistemas, é possível atrair também empresas que já utilizem outros sistemas para este efeito.
- Utilização de fontes externas que permitam obter este tipo de dados – Isto permite a existência prévia de alguns dados no sistema quando for inicializado, não havendo dependência de adição de dados diretamente por utilizadores, e uma maior variedade destes devido à obtenção de várias fontes (e.g. dados do Curriculum extraídos através do LinkedIn ou CV Europass)

3.2. Público-alvo

Como já referido anteriormente, os principais utilizadores desta plataforma serão:

- Candidatos: Pessoas que estejam interessadas em consultar informação sobre ofertas de emprego, sendo estes a principal fonte de perfis de candidatos.
- Empresas: Empresas que tencionam encontrar candidatos para as ofertas de emprego que possuem, e das quais serão a principal fonte.

Note-se que para este componente que estamos a levantar requisitos (Joobian cliente móvel) iremos nos focar nos Candidatos.

Tendo em conta que cada um destes adiciona ao sistema os dados que interessam ao outro, há aqui a criação de um efeito de bola de neve que vai assegurar o incremento do valor da plataforma ao longo do tempo (**Erro! A origem da referência não foi encontrada.**).

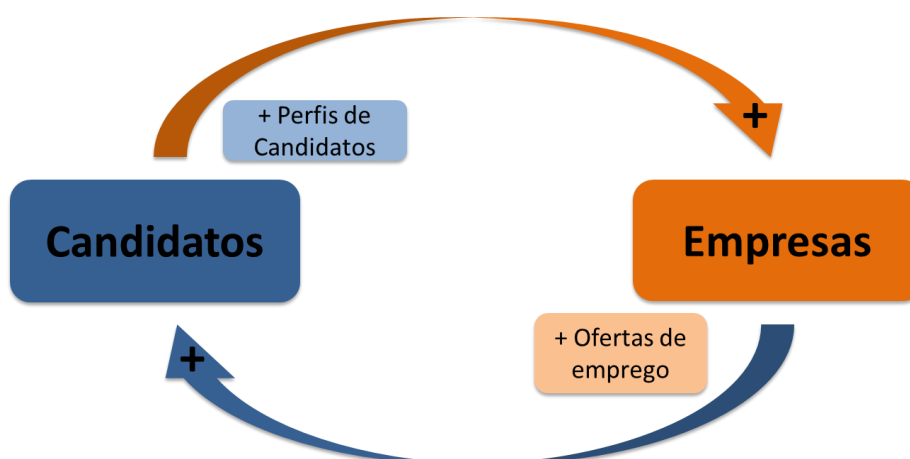


Figura 1 - Representação do efeito de bola de neve gerado

Numa segunda fase, tendo dados suficientes presentes no sistema, será possível fazer uma análise destes que permita oferecer aos seus utilizadores um serviço mais direcionado às suas necessidades através de recomendações e métricas obtidas da análise. Isto irá permitir uma correspondência entre candidatos e ofertas de emprego.

3.3. Plataforma JOOBIAN

A plataforma Joobian irá fornecer serviços de apoio ao cliente móvel que iremos desenvolver e detalhar ao longo deste documento, sendo que para entender melhor que é a plataforma Joobian iremos dar uma visão geral desta plataforma de recrutamento que terá então (conforme descrito no capítulo anterior) que ser capaz de interagir e obter dados de várias fontes externas.



Figura 2 - Vista global da plataforma e conexões com fontes externas.

Como podemos identificar na figura acima, há quatro fontes que poderão ser utilizadas para introdução direta ou indireta de dados no sistema.

- Portais de emprego: Estes serão utilizados para obter as últimas ofertas publicadas e possíveis de obtenção por feeds.
- Currículos Europass: Para facilitar o registo dos dados curriculares do utilizador, este poderá utilizar um currículo, formato Europass, para preencher campos do formulário de registo.
- LinkedIn: Em vez do currículo, descrito em cima, um utilizador pode também utilizar a informação do seu perfil do LinkedIn para o mesmo efeito, pois este funciona como um currículo *online*.
- Outras Plataformas: Pode também haver integração com outras plataformas de modo a permitir que empresas que já utilizem outros sistemas possam utilizar serviços disponibilizados pela nossa plataforma.

A vantagem de ter duas fontes para obter informação para preenchimento de dados do utilizador é a existência de possíveis utilizadores que podem ter apenas uma das duas, assim a probabilidade de um destes conseguir importar os dados de outro sítio onde já estejam preenchidos é maior.

Feeds RSS:

Os portais de emprego identificados que poderão ser utilizados como fontes de feeds foram o Sapo emprego, o Expresso emprego e o ITJobs. Como estes feeds devolvem um ficheiro XML e para cada uma das fontes o esquema é diferente, é necessário considerar a utilização de uma configuração para cada fonte assim como a possibilidade de

alteração do esquema por questões de manutenção e adição de novas fontes. Para isto deverá ser possível configurar genericamente qualquer fonte de feeds que se pretenda utilizar.

Como vão sendo lançadas novas ofertas de emprego nos feeds, esta extração deve ser feita periodicamente de maneira a que estas ofertas vão sendo adicionadas à plataforma.

Currículos Europass:

Estes vão ser utilizados na altura do registo ou edição de um perfil de utilizador para ajuda no preenchimento do formulário do perfil. É necessário que o utilizador carregue o ficheiro para a plataforma, que terá que ser em formato PDF. Depois de ser carregado o sistema terá que ler o mesmo e preencher os campos do formulário com a informação que for possível extrair.

É necessário considerar as várias versões existentes do formato Europass, pois cada versão tem um esquema e modelo de dados diferente e tem que ser tratada de maneira diferente, assim como as várias línguas. Também aqui terá que haver uma configuração que permita definir isto.

LinkedIn:

Os dados de perfil do LinkedIn, tal como os currículos, podem ser utilizados no registo e edição de um perfil de candidato. Este acesso é feito através da API do LinkedIn. Para isto é necessário ter uma aplicação registada no site de programador do LinkedIn¹, pois é preciso as credenciais obtidas neste registo para aceder autenticar o acesso à API. É ainda necessário que o utilizador que queira utilizar este método de extração conceda autorização para tal, o que resulta na obtenção de um *access token* que é utilizado para aceder aos dados do perfil do mesmo.

Como após ter o *access token* do utilizador é possível aceder à informação do seu perfil posteriormente, a não ser que expire a sua validade ou o utilizador termine a autorização dada, é pretendido fazer uma atualização periódica destes dados de modo a que o utilizador, ao atualizar o seu perfil no LinkedIn, este seja automaticamente atualizado na plataforma.

Outras Plataformas:

Para questões de integração do sistema com outras plataformas que os clientes utilizem, será disponibilizado uma API sob a forma de um serviço web RESTful. Este serviço devolverá respostas em JSON e permitirá que os clientes da plataforma tenham ao seu dispor as mesmas funcionalidades que teriam acedendo à aplicação web, a não ser que tal não seja possível por questões de visualização.

¹ <https://developer.linkedin.com/>

Além destas funcionalidades descritas o sistema deve disponibilizar consulta de ofertas de emprego e perfis de candidatos através de pesquisa, por palavras-chave e filtros, e recomendações geradas através de análise de dados.

Este deve também ter em conta a possível adição de mais módulos no futuro que possibilitem integração com outros tipos de sistemas, por isso deverá seguir uma arquitetura modular.

4. Casos de Teste

4.1. Acesso

ID do Caso de Teste	Sign in
<p><i>Pré-Requisitos</i></p>	<p>JOOB.A_SignIn.01v1</p> <p>O utilizador deverá ter disponíveis os seguintes dispositivos móveis (emuladores configurados): Android – Versão 4.4.2 resolução 720x1080 iOS – Versão 7.1.1 aspect ratio 40/71</p> <p>O utilizador deverá ter a aplicação JOObian instalada no dispositivo móvel (emulador).</p> <p>O utilizador deverá ter acesso à internet.</p> <p>O utilizador já deverá estar previamente registado na plataforma (com credenciais de acesso).</p>
<p><i>Descrição das Acções</i></p>	<ol style="list-style-type: none"> 1. Aceder à aplicação JOObian – beta; 2. Clicar no botão “Sign in”; 3. No campo “Email” introduzir o email de conta (credenciais de acesso do utilizador); 4. No campo “Password” introduzir a password de conta (credenciais de acesso do utilizador); 5. Clicar no botão “Done”.
<p><i>Pós-Requisitos</i></p>	<p>Deverá ser garantido o correcto funcionamento da aplicação para as respectivas versões/resoluções dos dispositivos:</p> <ul style="list-style-type: none"> • Android – Versões 4.0.4, 4.1, 4.2.2, 4.3. Resoluções 320x480, 480x800, 540x960, 600x905, 720x1080, 800x1280, 1080x1920 • iOS – Versões 6.1, 7.0, 7.1. Aspect Ratios 2/3, 3/4, 40/71
<i>Critérios de Aceitação</i>	<i>Critérios de Recomeço</i>
<p>Após o passo 1, deverão ser apresentados ao utilizador dois botões:</p> <ul style="list-style-type: none"> • “Sign up for JOObian”; • “Sign in”. <p>Após o passo 2, deverá ser apresentado ao utilizador o botão de acesso por intermédio das credenciais de acesso ao “Linkedin” (botão “Log in with Linkedin”).</p> <p>Deverão ser apresentados os campos de texto: “Email” e “Password”.</p> <p>Deverão ser apresentados os botões de: “Cancel” e “Done”.</p> <p>Deverá ser apresentado o link “I forgot my password”.</p>	<p><i>Critérios de Suspensão</i></p> <p>Nenhum.</p> <p>Repetir o caso de teste, mas no passo 5 clicar no botão “Cancel”. Após este passo, o utilizador deverá ser reencaminhado para a janela do passo 2.</p> <p>Repetir o caso de teste, mas no passo 3 e 4 colocar credenciais de acesso erradas. Após este passo, deverá ser apresentada a mensagem de alerta “No account this email. Please Sign up for JOObian”. O utilizador deverá ser</p>

Após o passo 5, o utilizador deverá ter acesso às funcionalidades da aplicação.	<p>redireccionado para a janela de registo "Sign up for JOObian".</p> <p>Repetir o caso de teste, mas saltar o passo 3. Após este passo, deverá ser apresentada a mensagem de alerta "Email address needed!".</p> <p>Repetir o caso de teste, mas saltar o passo 4. Após este passo, deverá ser apresentada a mensagem de alerta "Password needed!".</p>	
---	--	--

4.2. Settings

ID do Caso de Teste	JOOB.S_CJO.01v1	Countries for Jobs Offers		
Pré-Requisitos	<p>O utilizador deverá ter disponíveis os seguintes dispositivos móveis (emuladores configurados): Android – Versão 4.4.2 resolução 720x1080 iOS – Versão 7.1.1 aspect ratio 40/71</p> <p>O utilizador deverá ter a aplicação JOObian instalada no dispositivo móvel (emulador).</p> <p>O utilizador já deverá estar previamente registado na plataforma (com credenciais de acesso).</p> <p>Ter na opção “Countries for Job Offers” todos os países seleccionados (Portugal, Brazil, United Kingdom, United States).</p>			
Descrição das Acções	<ol style="list-style-type: none"> 1. Aceder à aplicação JOObian – beta (credenciação efectuada); 2. Clicar no botão “Settings”; 3. Clicar no botão opção “Countries for job offers”; 4. Deixar apenas a opção “Portugal” seleccionada; 5. Clicar no botão “Job Offers”. 			
Pós-Requisitos	<p>Deverá ser garantido o correcto funcionamento da aplicação para as respectivas versões/resoluções dos dispositivos:</p> <ul style="list-style-type: none"> • Android – Versões 4.0.4, 4.1, 4.2.2, 4.3. Resoluções 320x480, 480x800, 540x960, 600x905, 720x1080, 800x1280, 1080x1920 • iOS – Versões 6.1, 7.0, 7.1. Aspect Ratios 2/3, 3/4, 40/71 			
Critérios de Aceitação		Critérios de Recomeço		Critérios de Suspensão
Após o passo 2, na opção “Countries for job offers” deverão estar visíveis os nomes		Repetir o caso de teste, mas no passo 4 deixar apenas a		Nenhum.

<p>dos países seleccionados (neste caso deverão estar "Portugal, Brazil, United Kingdom, United States").</p> <p>Após o passo 3, na opção "Countries for job offers" deverão estar todos os países seleccionados "Portugal, Brazil, United Kingdom, United States".</p> <p>Nota1: deverá estar visível a mensagem "At least one country must be selected".</p> <p>Nota2: deverá ser impossível desseleccionar todas as opções no passo 4.</p> <p>Após o passo 5, todas as "job offers" apresentadas na listagem deverão ser respeitantes a "Portugal".</p>	<p>opção "Brazil" seleccionada. Após o passo 5, todas as "job offers" apresentadas na listagem deverão ser respeitantes ao "Brazil".</p> <p>Repetir o caso de teste, mas no passo 4 deixar apenas a opção "United Kingdom" seleccionada. Após o passo 5, todas as "job offers" apresentadas na listagem deverão ser respeitantes ao "United Kingdom".</p> <p>Repetir o caso de teste, mas no passo 4 deixar apenas a opção "United States" seleccionada. Após o passo 5, todas as "job offers" apresentadas na listagem deverão ser respeitantes aos "United States".</p> <p>Repetir o caso de teste, mas no passo 4 deixar todas as opções "Portugal, Brazil, United Kingdom, United States" seleccionadas. Após o passo 5, todas as "job offers" apresentadas na listagem deverão ser respeitantes aos países "Portugal, Brazil, United Kingdom, United States".</p>	
--	--	--